APOSTILA SCRATCH 2026

Apostila elaborada pelo Professor Pedro Pereira da Silva.

Professor **O**rientador de **E**ducação **D**igital da *Emef Prof. Josué de Castro.*

Caro (a) estudante, esta apostila traz uma série de atividades sobre programação em blocos utilizando a linguagem **Scratch** (versão 3.29.1).

Estudar programação pode trazer diversos benefícios para o seu desenvolvimento cognitivo, como por exemplo, o desenvolvimento do raciocínio lógico, a melhora na aprendizagem em Matemática, o lidar com frustações, além do estímulo a imaginação, autonomia e ao trabalho em equipe.

Bem vindo (a) ao mundo da programação ☺

Professor Pedro Pereira da Silva (desenvolvedor)

email: pedropps@alumni.usp.br

instagram: @tec.assistiva.pedro

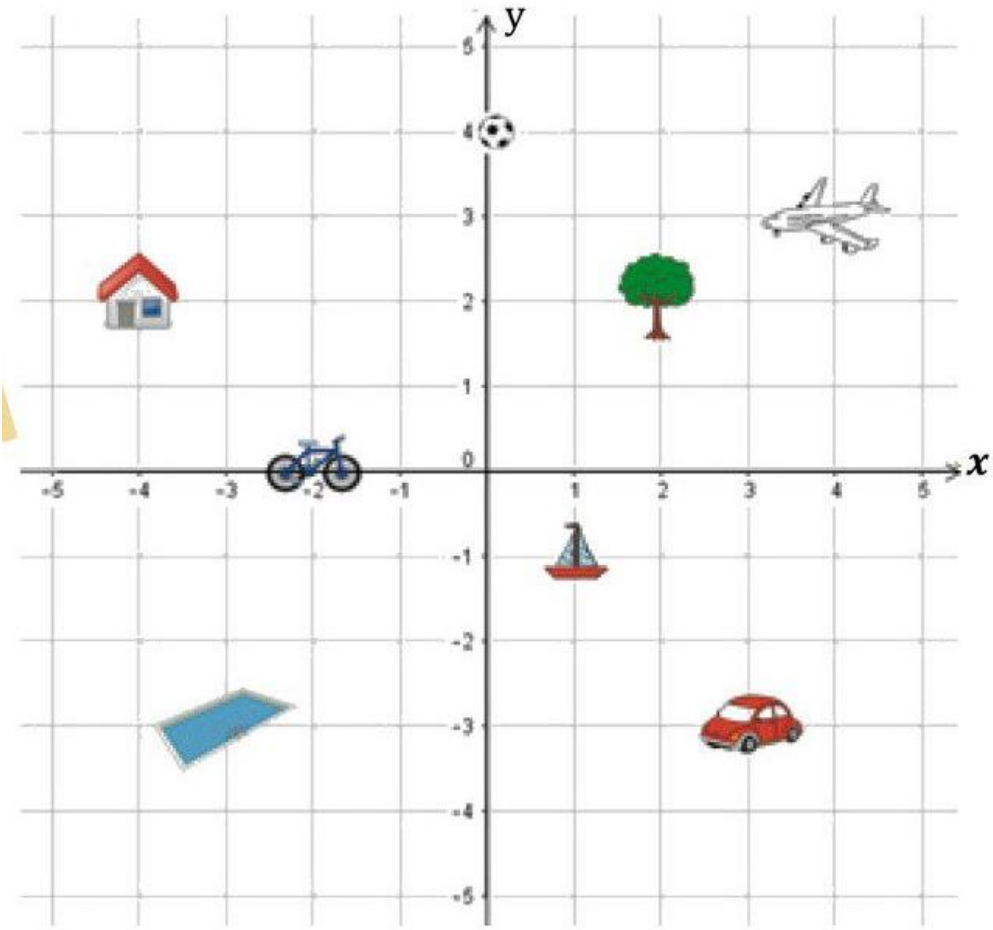
Professor Orientador de Educação Digital da Emef Professor Josué de Castro

Aula 1: Posicionando personagens na tela

Para posicionarmos personagens na tela, precisamos entender o conceito de **localização de objetos por coordenadas**. Em Matemática utilizamos a **letra *x*** para indicar o deslocamento horizontal ⟺ e a **letra *y*** para o deslocamento vertical ⇕.

Aqui é importante destacar que também utilizamos números negativos indicar deslocamentos à esquerda ou abaixo do ponto central do plano.

Exemplo:



Por convenção escrevemos as coordenadas na ordem (x , y).

Casa x=- 4 e y= 2

Avião x= 4 e y= 3

Árvore x= 2 e y= 2

Porta x= -3 e y= -3

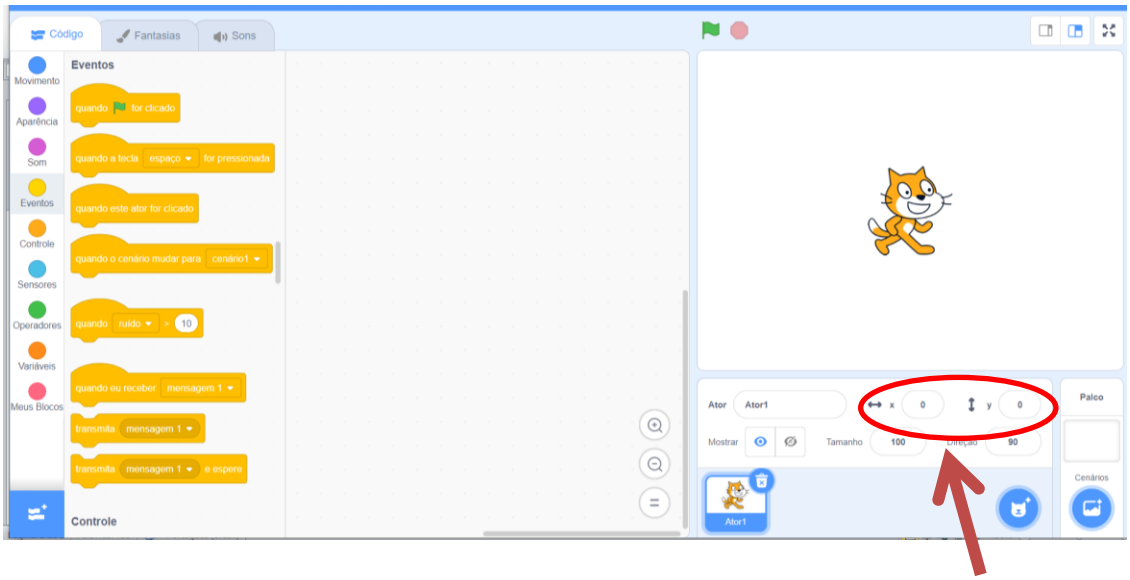
*Bola x= 0 e y= 4

*Bicicleta x= -2 e y= 0

Barco x=1 e y= -1

Carro x= 3 e y= -3

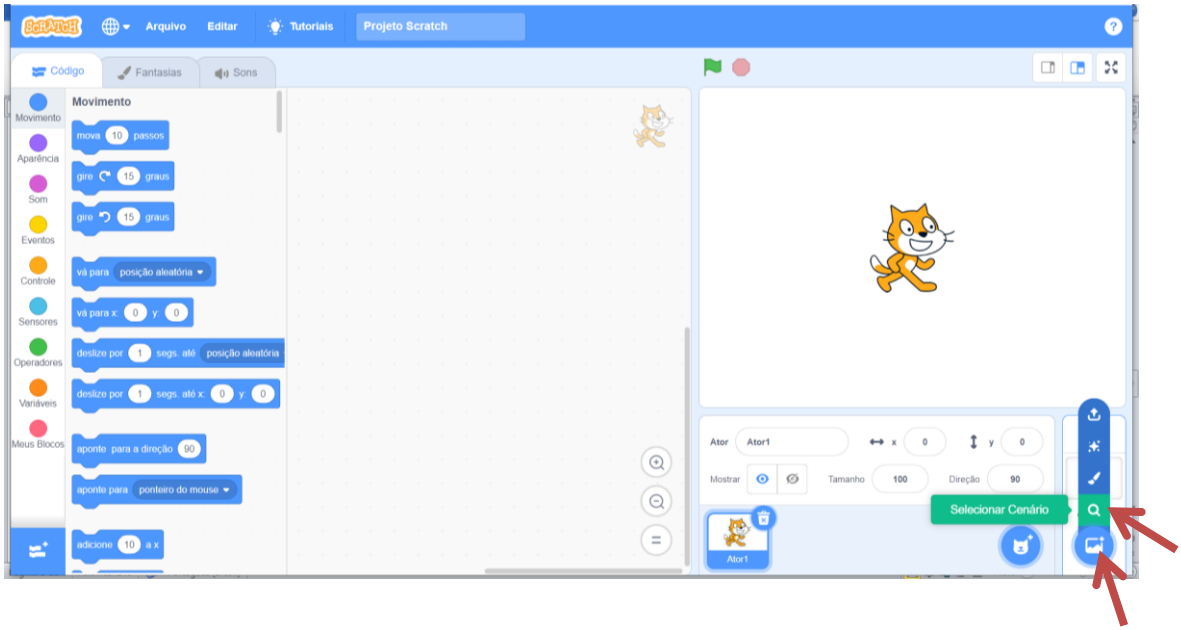
Atividade prática: Abra o Scratch no computador.



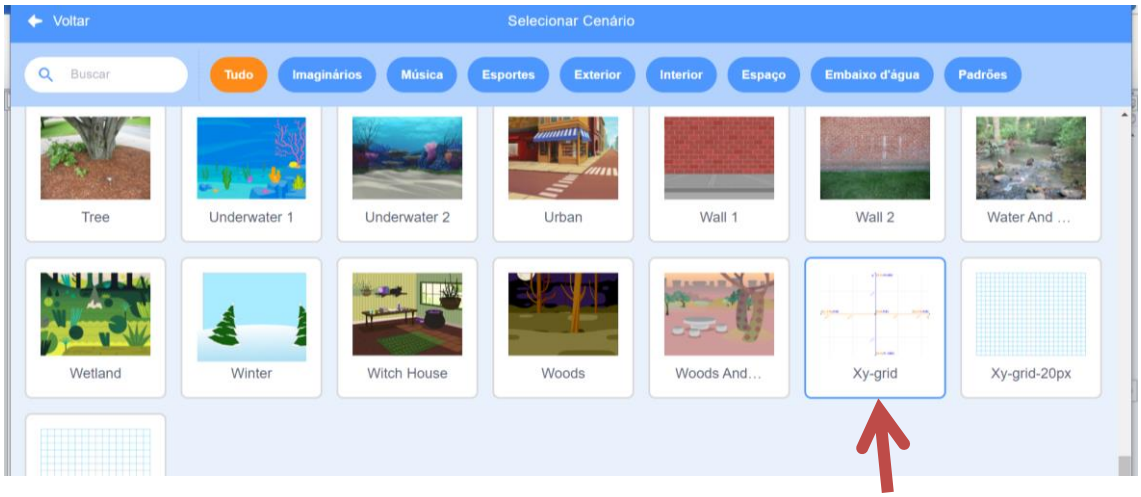
Por padrão, assim que abre o Scratch, o personagem já fica posicionado no **centro do plano com x: 0 e y: 0.**

A atividade consiste em movimentar o personagem na horizontal e vertical. Siga as orientações:

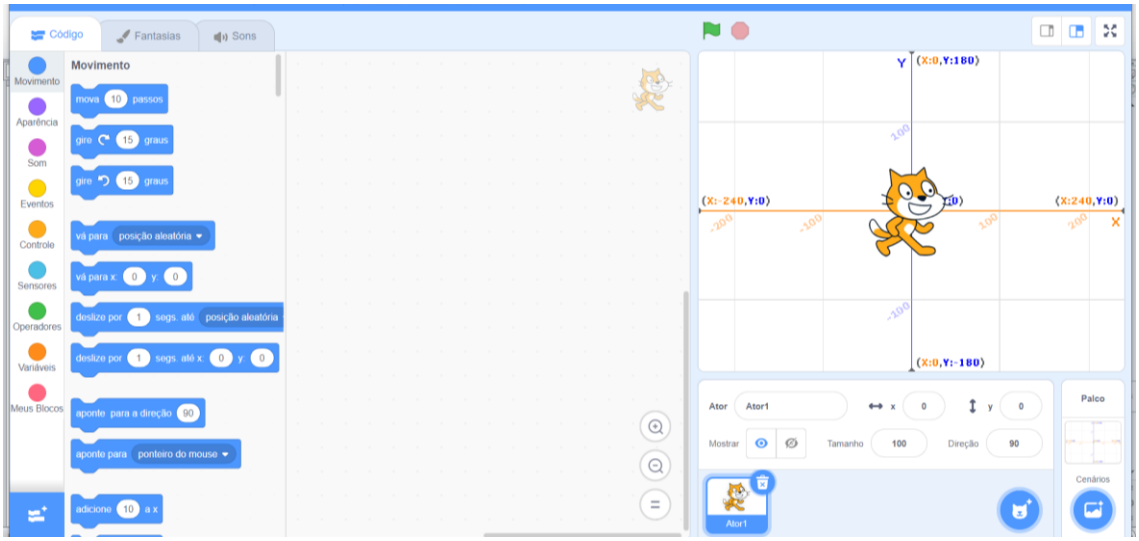
PASSO 1: Mova o cursor do mouse até "***cenários***" e clique na lupa:

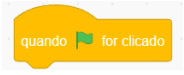


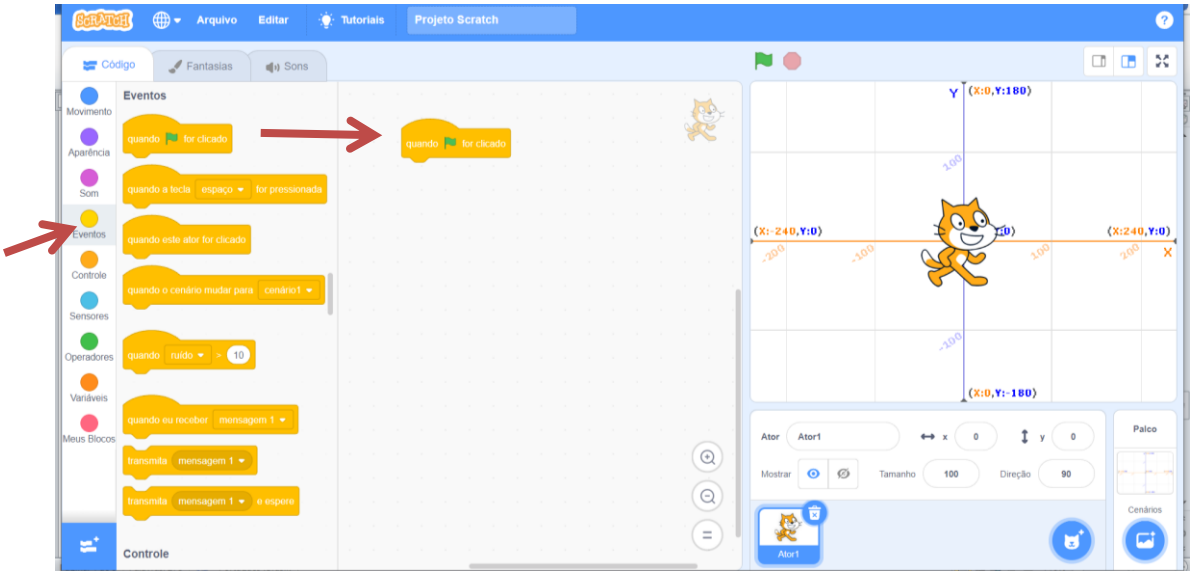
PASSO 2: Role a barra lateral e selecione o ***cenário Xy – grid***:



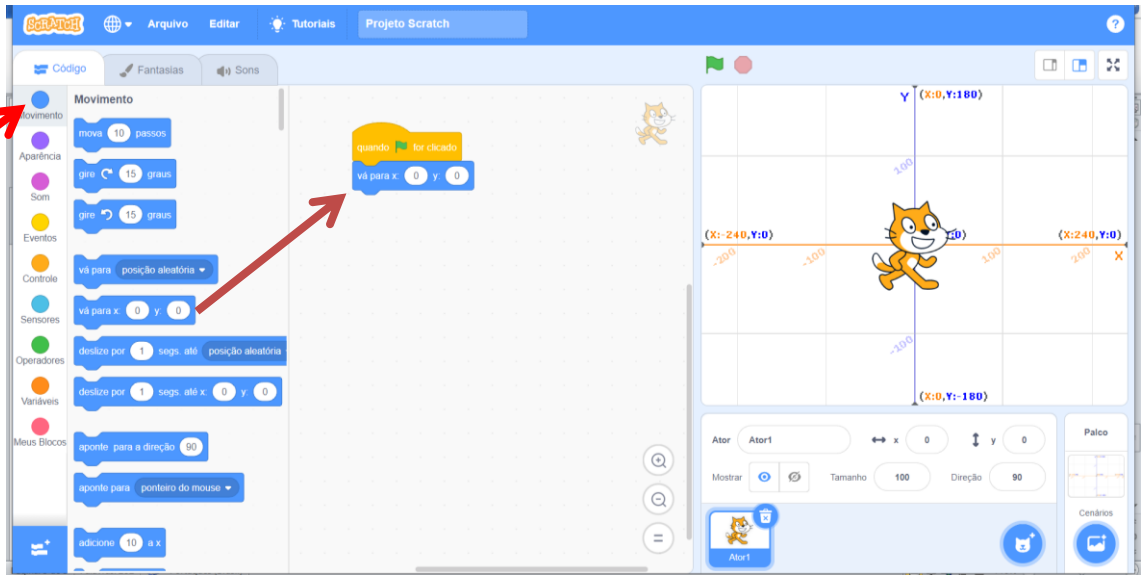
PASSO 3: Sua tela deve ficar assim:

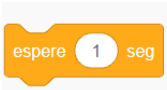


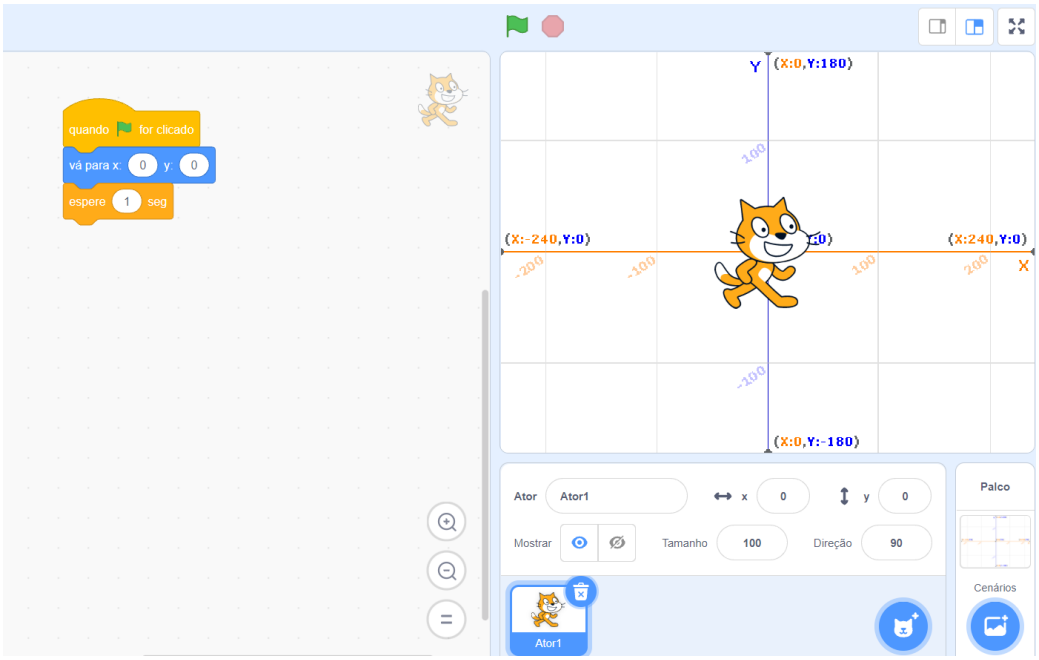
PASSO 4: Na barra de comandos do lado esquerdo, clique em ***Eventos*** e arraste  para o centro da tela.



PASSO 5: Agora clique em **Movimento** e arraste



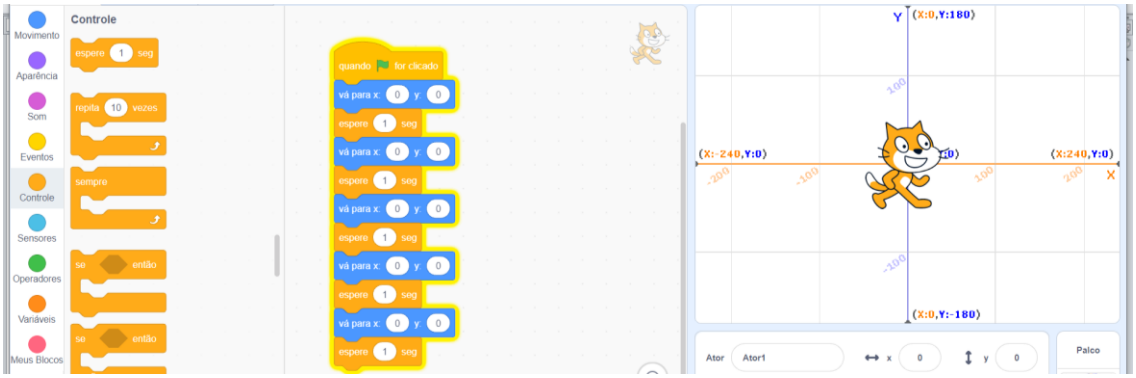
PASSO 6: Clique em **Controle** e arraste 



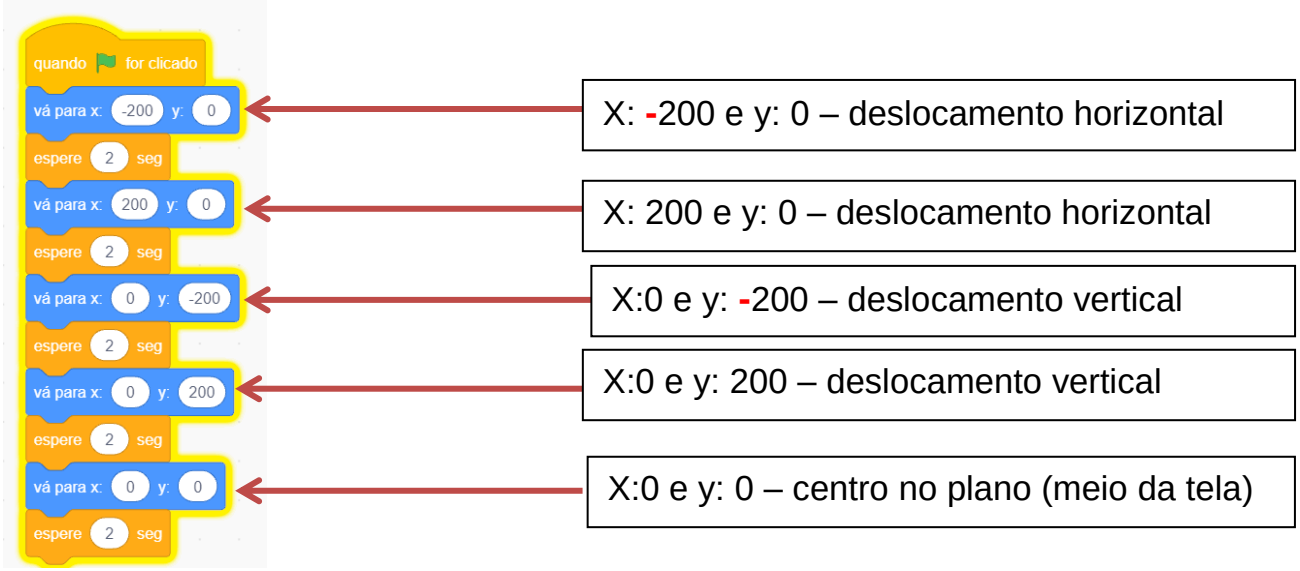
Vamos repetir esse bloco de comando por **mais quatro vezes**, entretanto, para facilitar, vamos usar um comando para **duplicar os blocos**. Essa ferramenta será muito útil no nosso curso.

Posicione o cursor do mouse sobre o bloco e clique com o **botão direito.** Isso vai habilitar a possibilidade de duplicar os blocos.

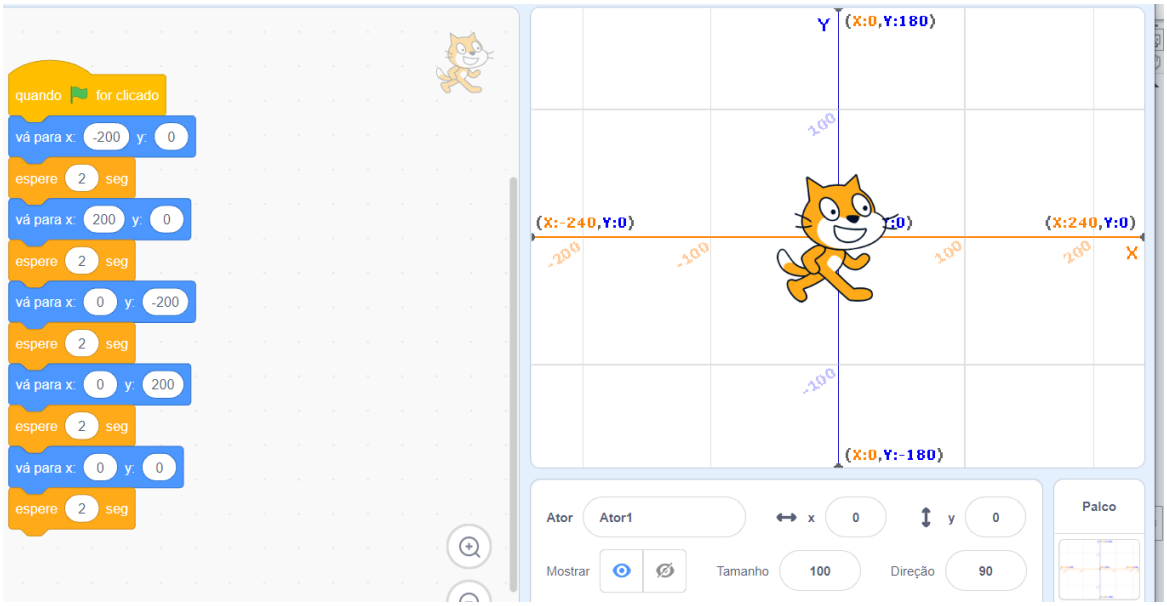
Escreva os blocos dessa forma:



Agora vamos escrever os valores de x e y nos blocos:



Agora basta realizar o teste, clique em . Se seu programa estiver correto, o personagem devo se deslocar na horizontal e depois na vertical.

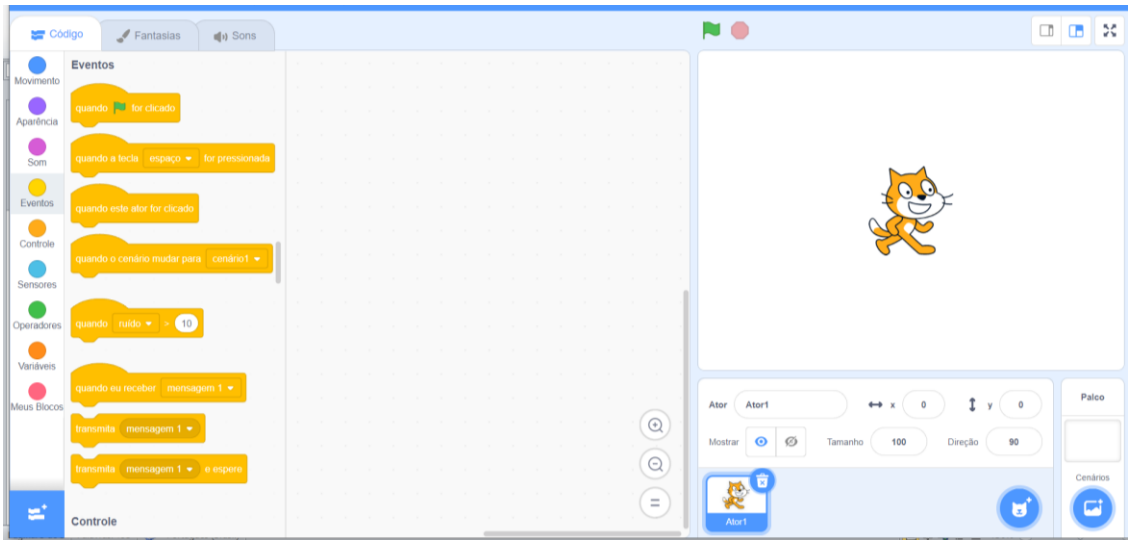


Parabéns! Você concluiu nossa primeira atividade do curso ☺

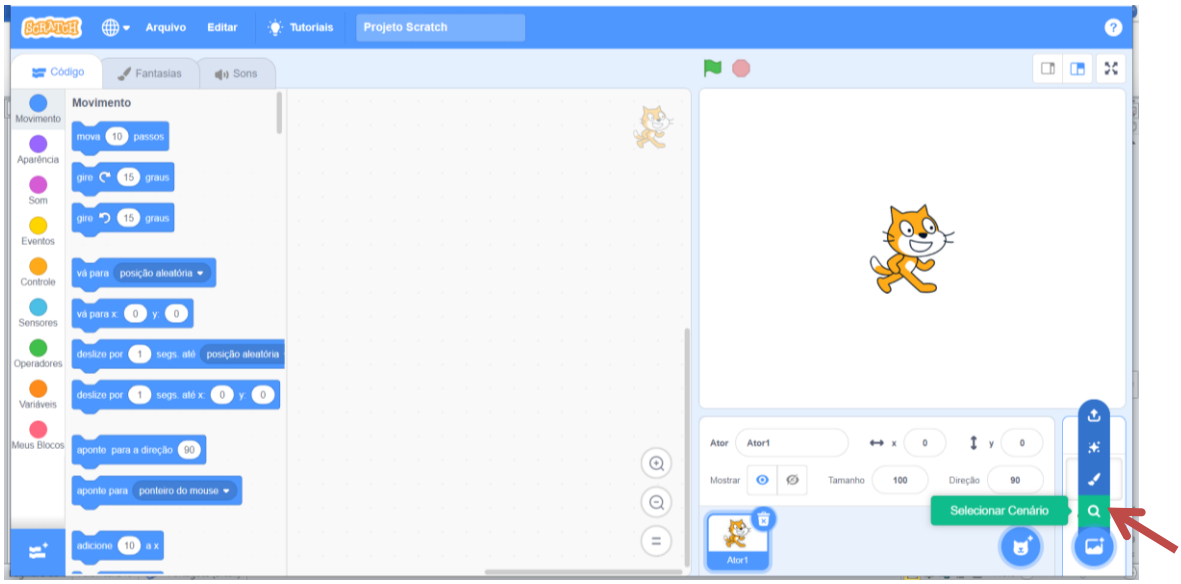
Aula 2: Deslocando o personagem no eixo horizontal

Nesta atividade prática, além do **deslocamento horizontal** (eixo x), vamos fazer com que o **personagem movimente as pernas** e dê uma mensagem através do **recurso do balão de fala**. Todo esse processo será feito **passo a passo**, ok?

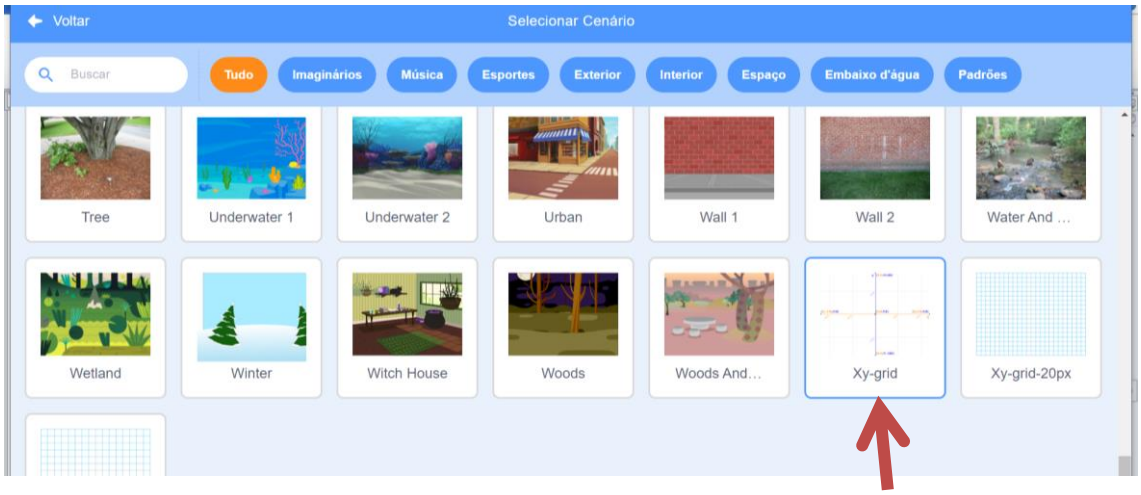
PASSO 1: Abra o Scratch



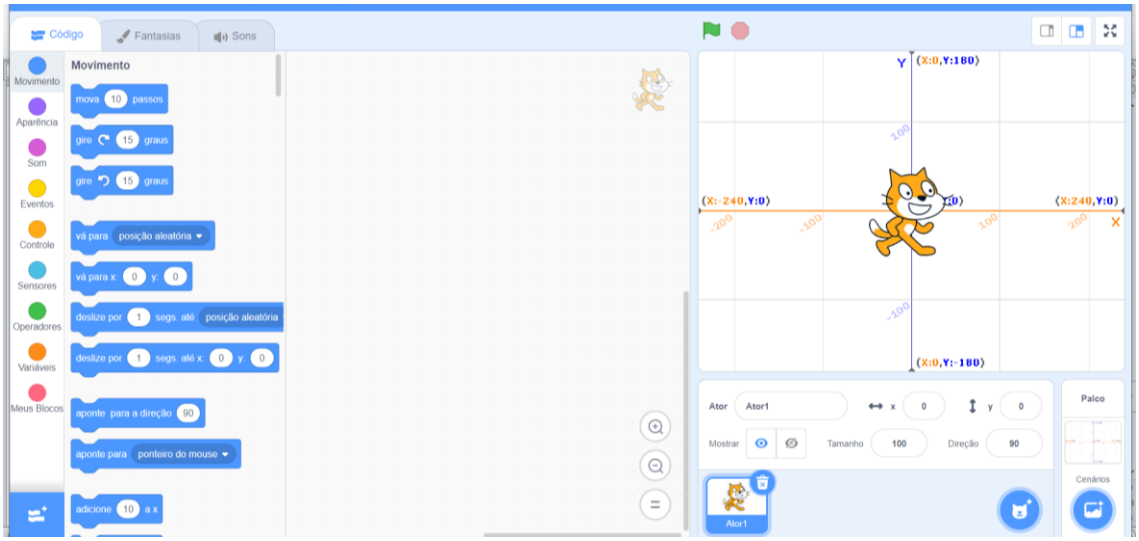
PASSO 2: Mova cursor do mouse até "***cenários***" e clique na lupa:



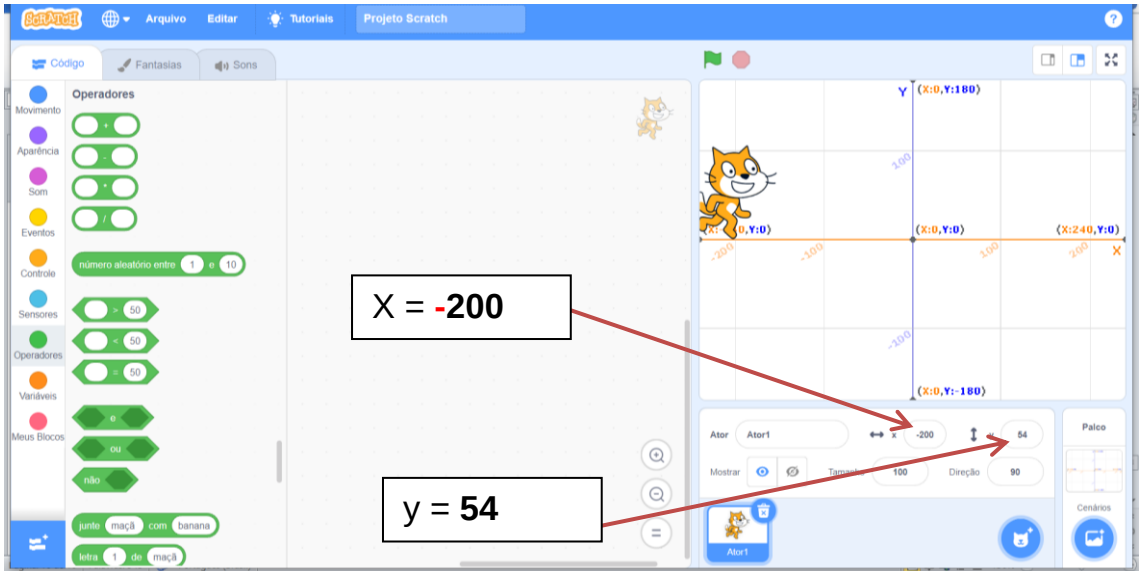
PASSO 3: Role a barra lateral e selecione o ***cenário Xy – grid***:



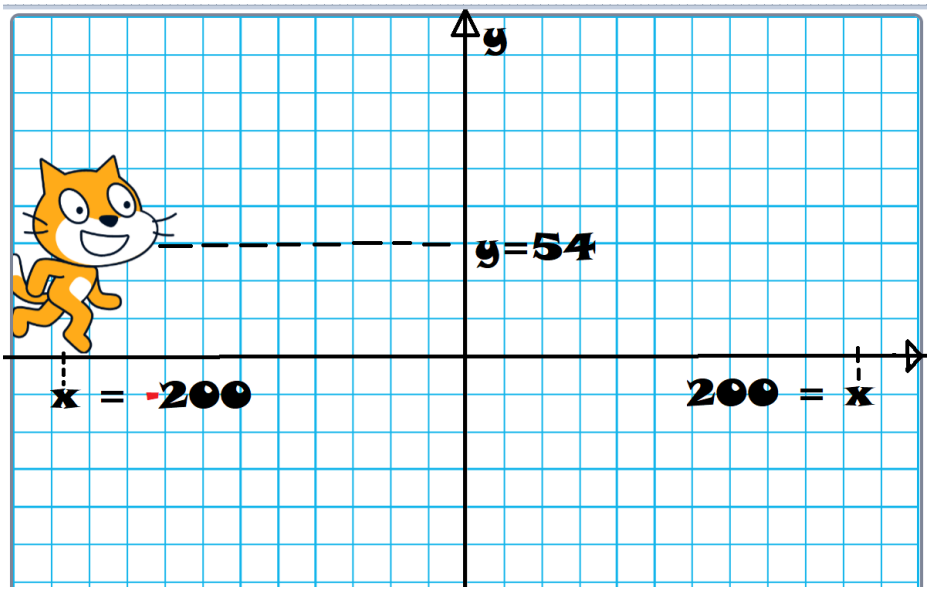
Sua tela deve ficar assim:



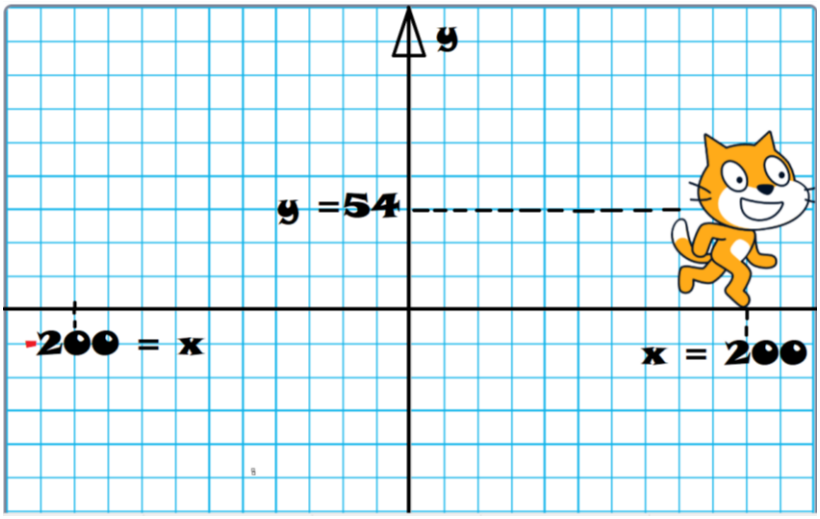
PASSO 4: Agora, vamos determinar as coordenadas x e y para que o personagem fique no lado esquerdo da tela. Escreva **x = -200** e **y=54** conforme indicado na imagem.



Observe que ao digitar os valores o personagem se deslocou para lado esquerdo da tela.

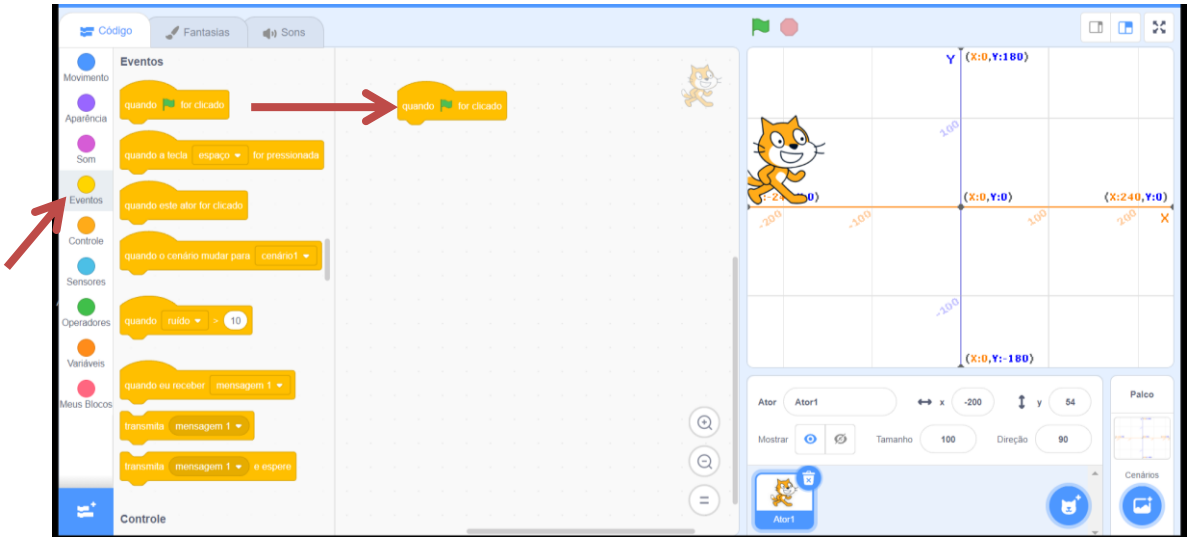


Agora, através da programação em blocos, vamos fazer com que ele se desloque no eixo horizontal para o lado direito, isto é, até a coordenada **x = 200 e y = 54.**



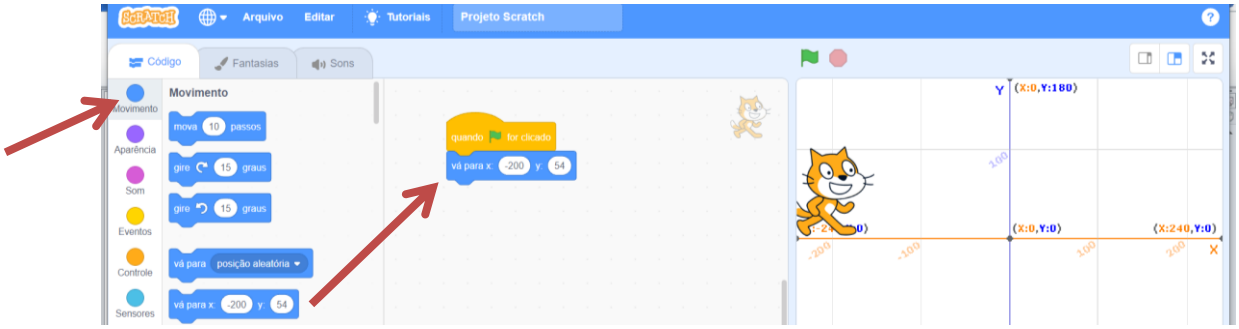
PASSO 5: Na barra de comandos do lado esquerdo, clique em "***Eventos"*** e arraste

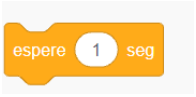
para o centro da tela.



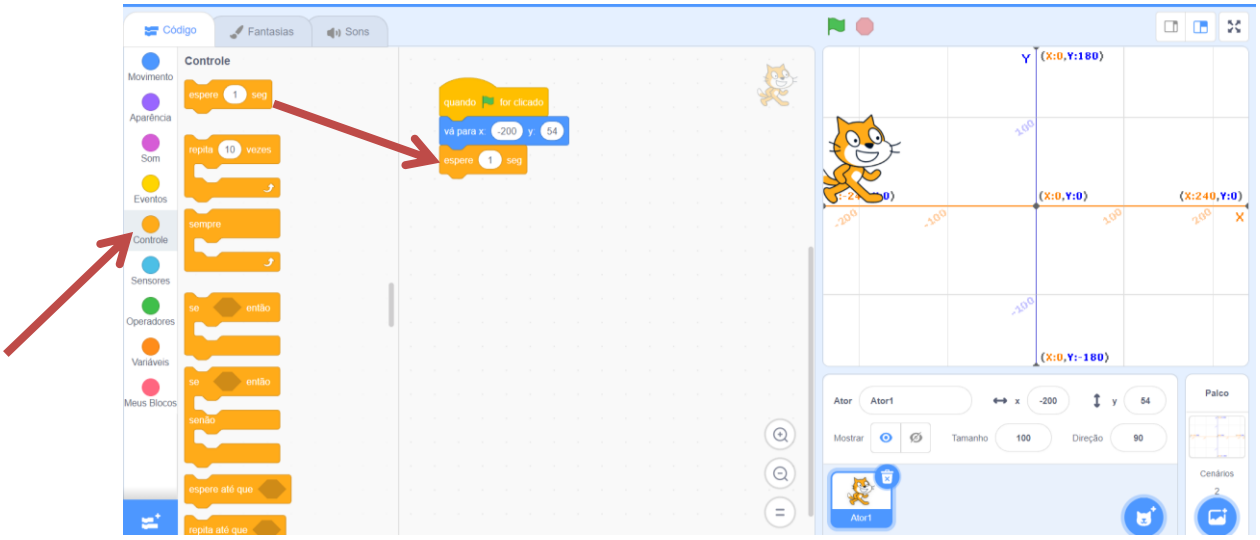
PASSO 6: Agora clique em "**Movimento"** e arraste

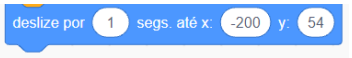
. Observe que os valores já estão preenchidos porque já o fizemos no início da programação.



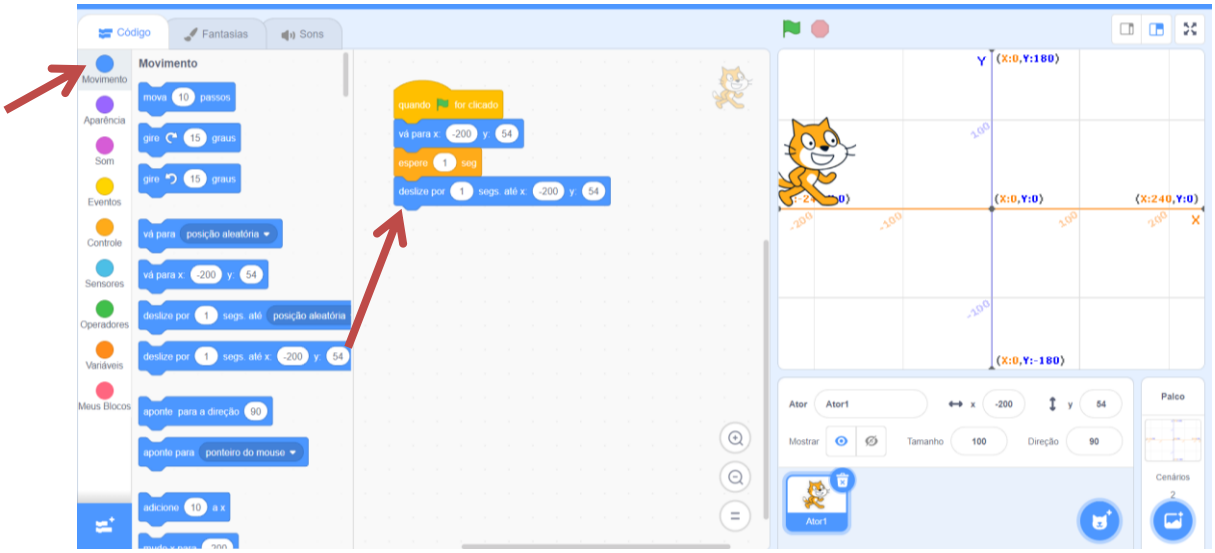
PASSO 7: Clique em “**Controle”** e arraste 

para o centro da tela.

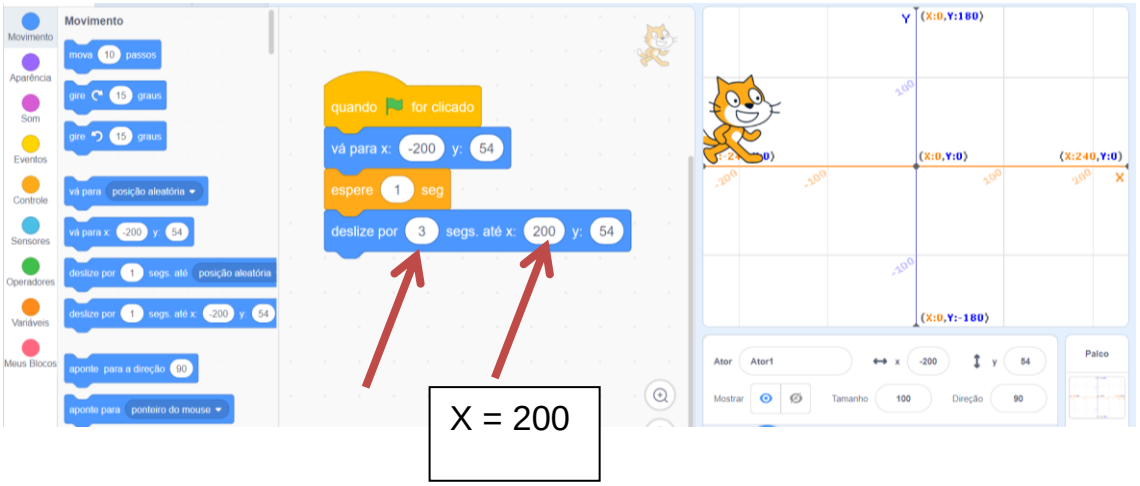


PASSO 8: Lembre-se que vamos deslocar o personagem até a posição **x = 200 e y = 54.** Clique em “Movimento” e arraste 

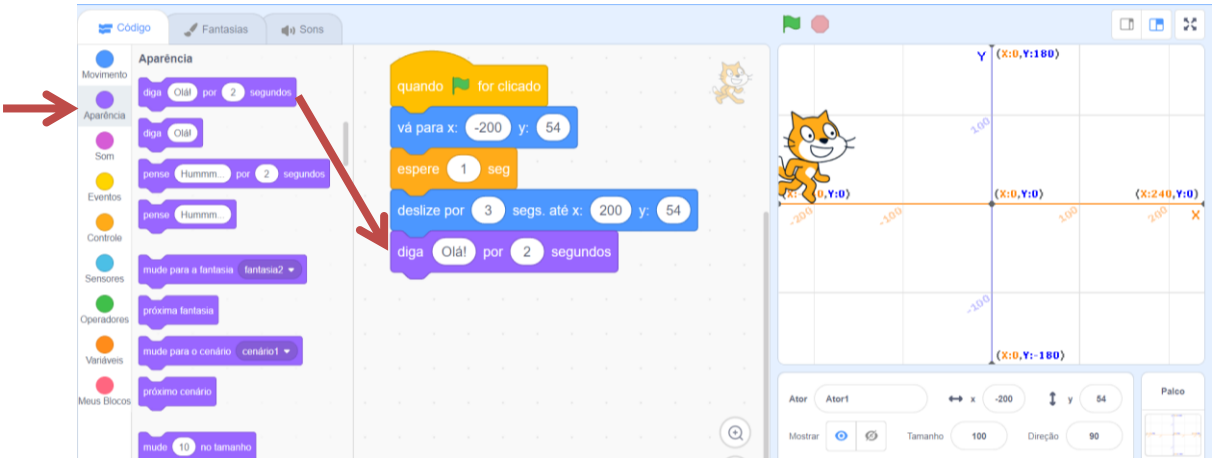
para o centro da tela.



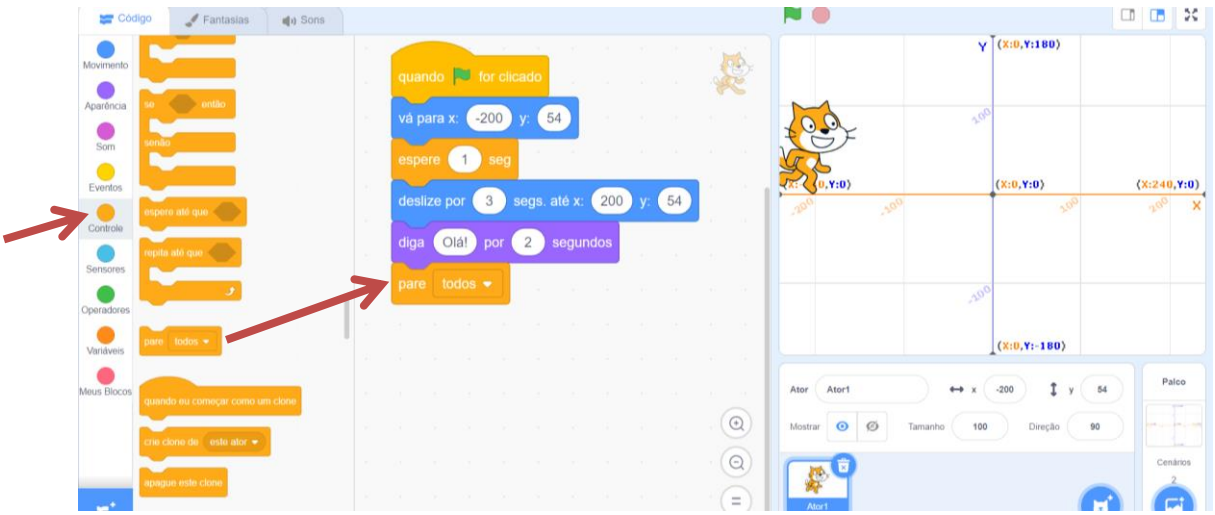
Observe que no comando os valores estão como x = **-** 200 (negativo) e y = 54, como queremos que se desloque pelo eixo horizontal, vamos alterar o valor de x **para x = 200** (positivo)**,** e o tempo vamos deixar em **3 segundos para deslizar.**



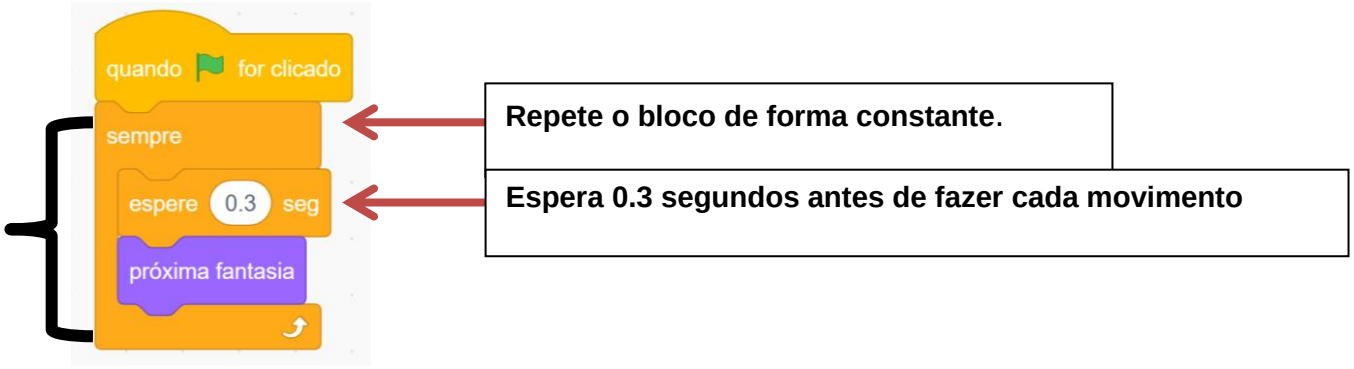
PASSO 9: Em **“Aparência”** arraste para nosso bloco de comando.



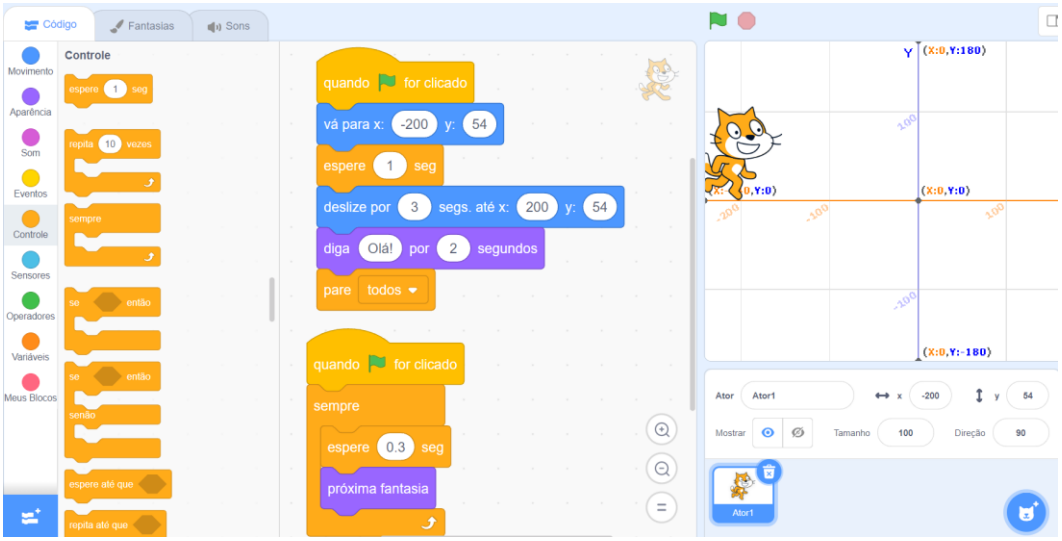
PASSO 10: Em **“Controle”** desça o cursor e arraste para nosso bloco de comando.



Passo 11: Para finalizar, vamos movimentar as pernas do personagem. Para movimentar personagens usamos a combinação dos comandos:



Seu bloco comando deve ficar assim:



Agora basta realizar o teste, clique em .

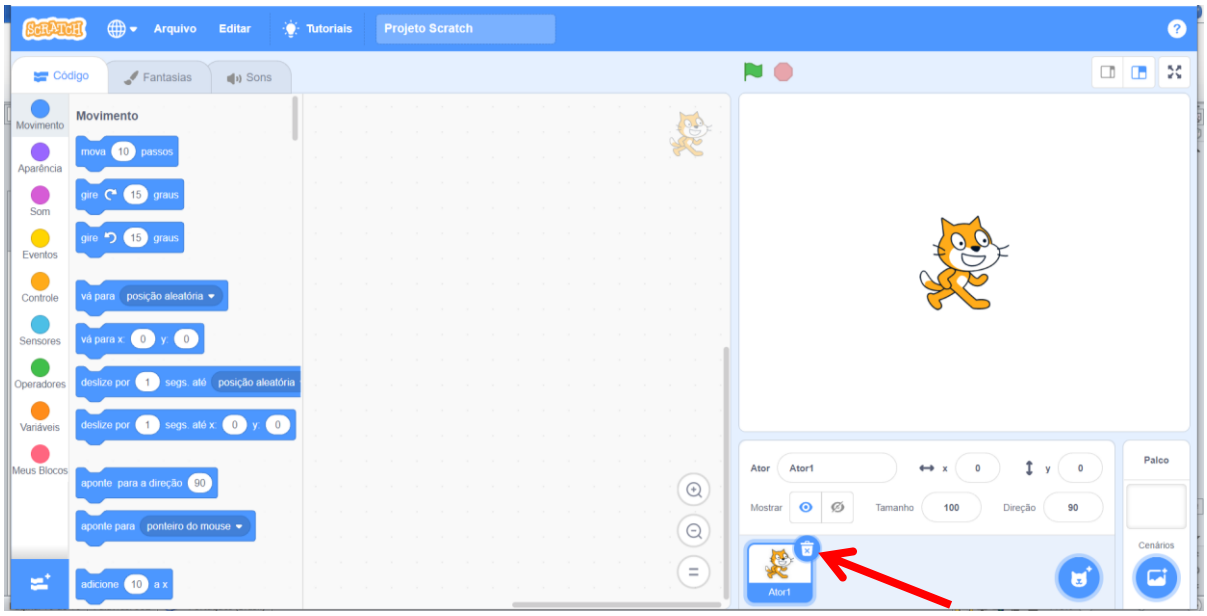
Parabéns! Você concluiu a segunda aula do nosso curso!

Aula 3: Nessa aula vamos criar uma história com dois personagens e dois cenários.

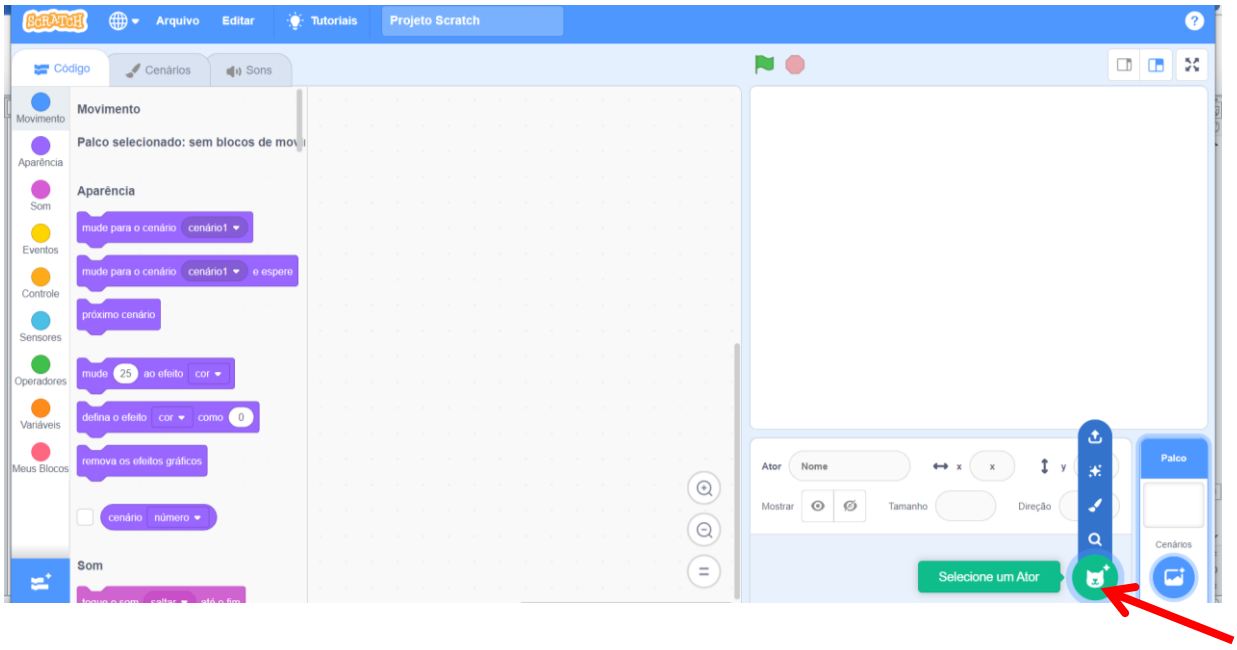
Um mago vai encontrar com um Elfo e fará um feitiço para que mudem de lugar.

Abre o Scratch e siga os passos:

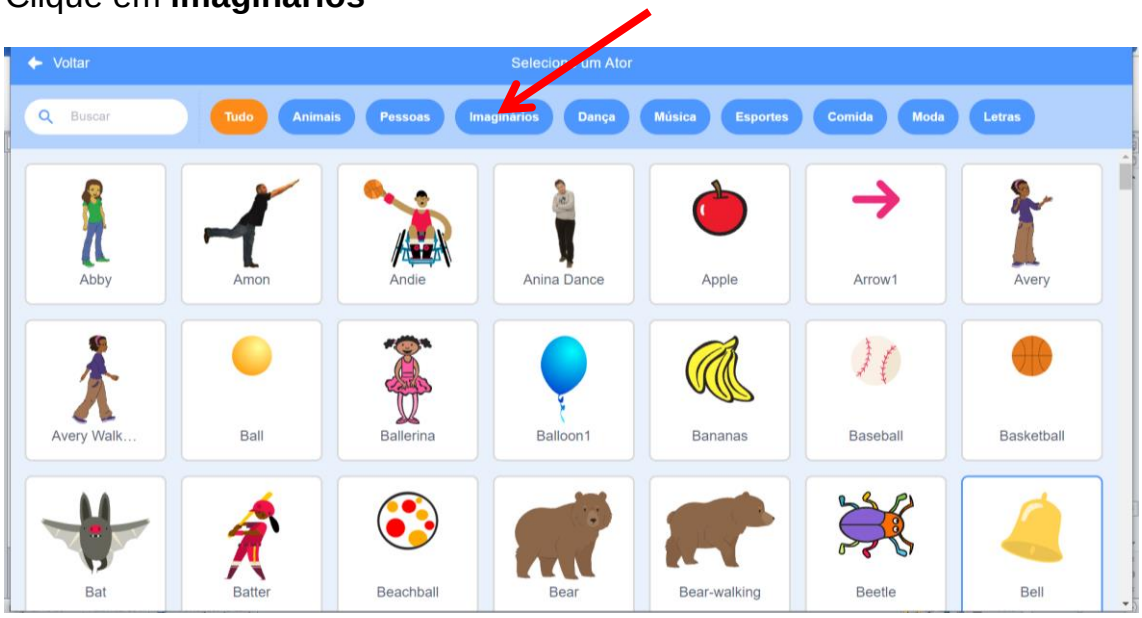
PASSO 1: Exclua o personagem ***ator 1***, clicando na ***lixeira*** que fica acima do ator.



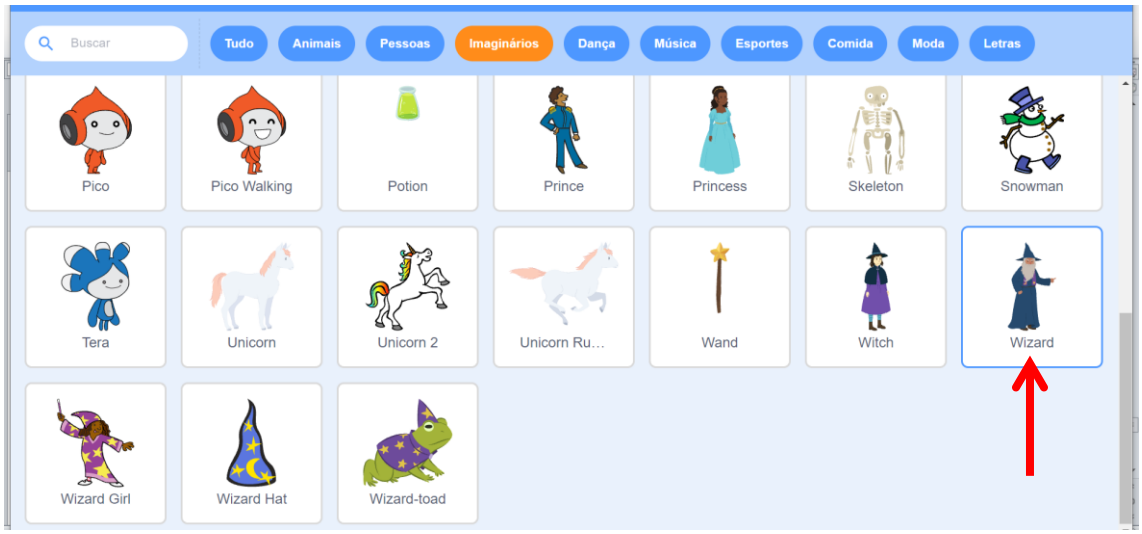
Clique em ***selecione um ator***



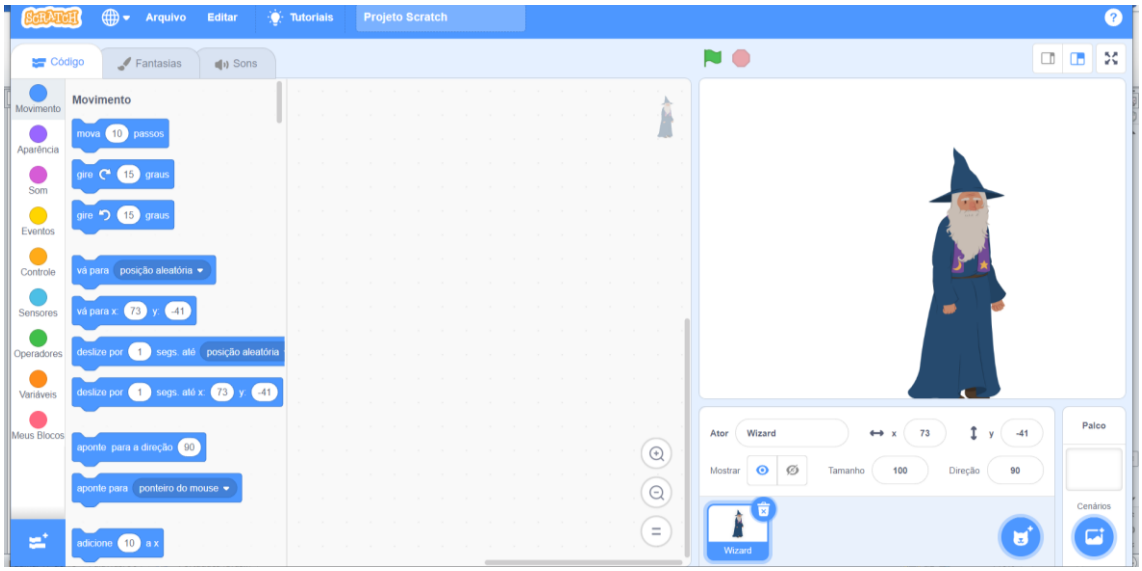
Clique em **imaginários**



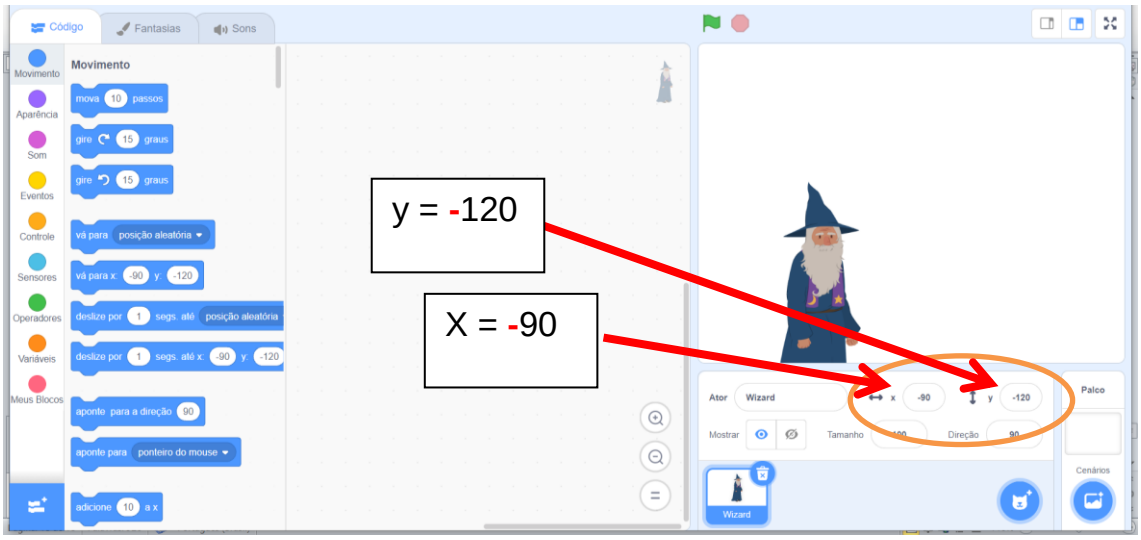
Mova o cursor do mouse até encontrar o ator ***Wizard***



Sua tela deve ficar assim:



Agora posicione o mago (wizard) com as coordenas x= -90 e y = -120



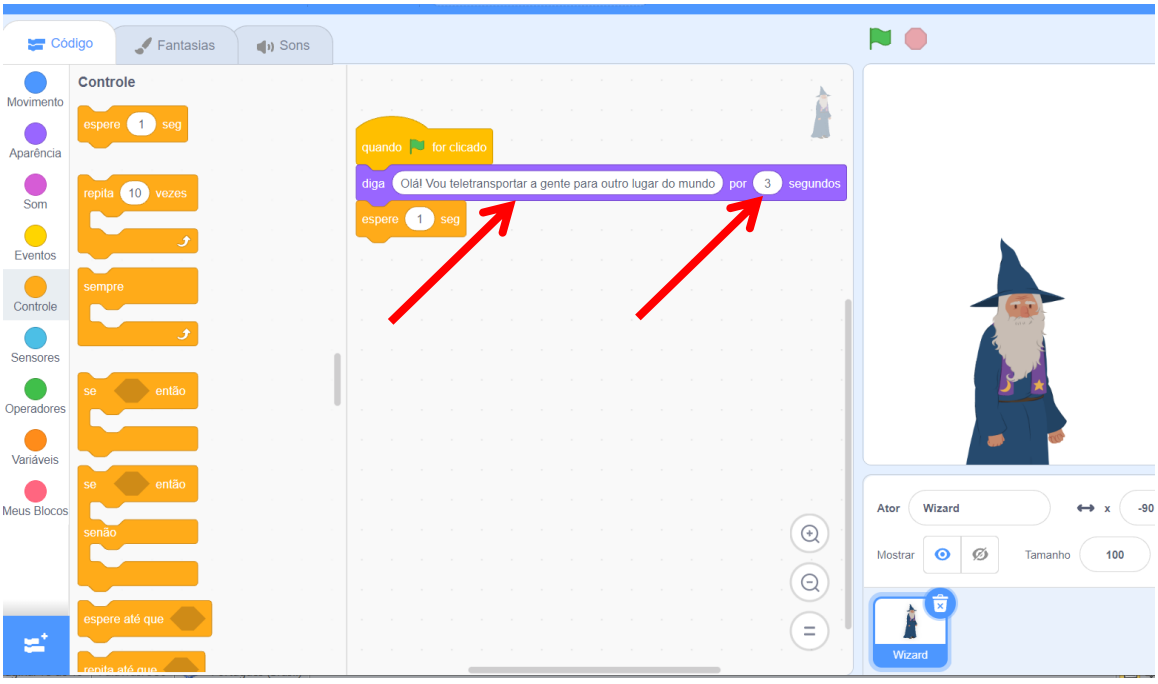
PASSO 2: Antes de acrescentarmos o segundo ator, vamos escrever a programação do mago.

Vamos combinar esses códigos:

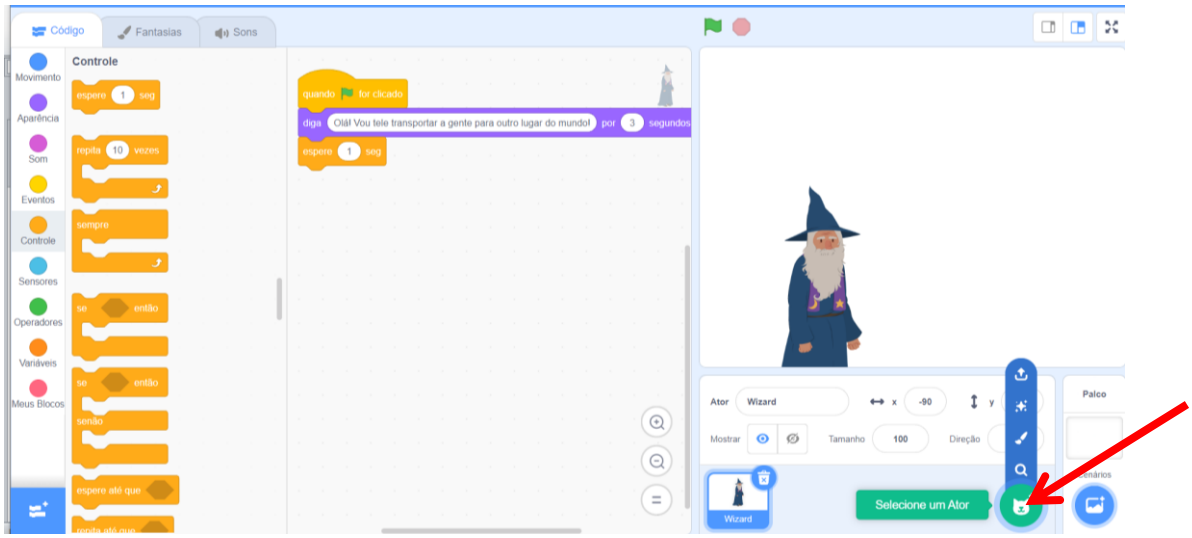
No comando

vamos alterar a ***"Olá!"*** por ***"Olá! Vou tele transportar a gente para outro lugar do mundo"***. E alterar o tempo para 3 segundos.

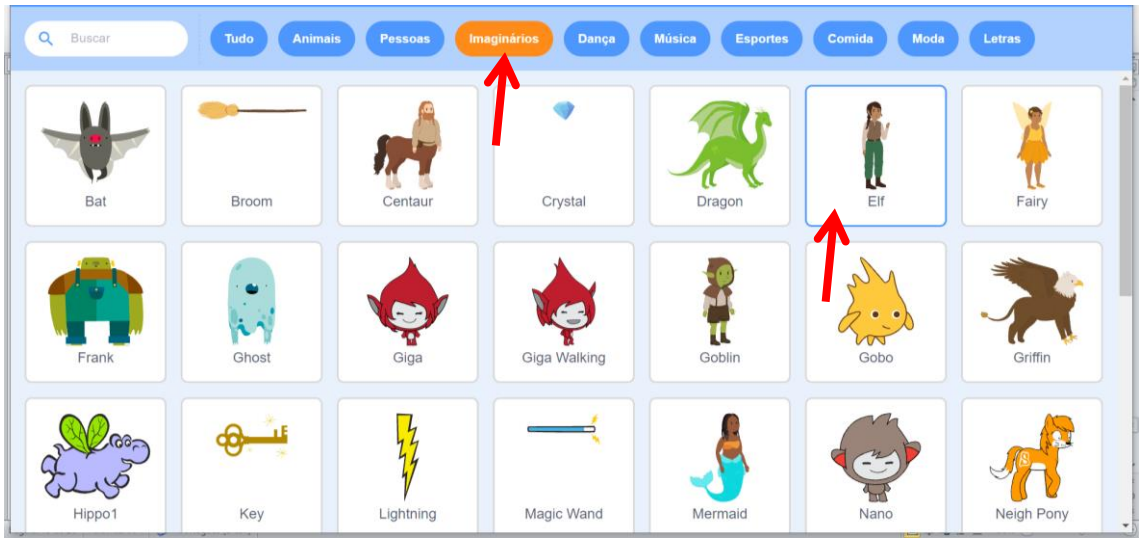
Sua tela deve ficar assim:



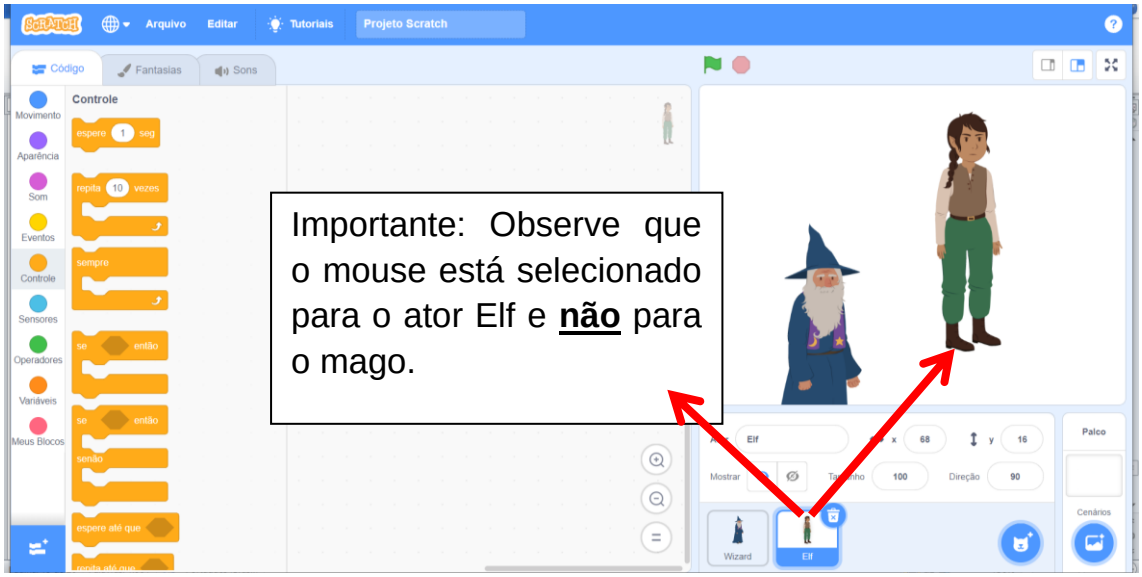
PASSO 3: Agora vamos acrescentar o segundo ator, ok? Clique me *selecionar um ator*



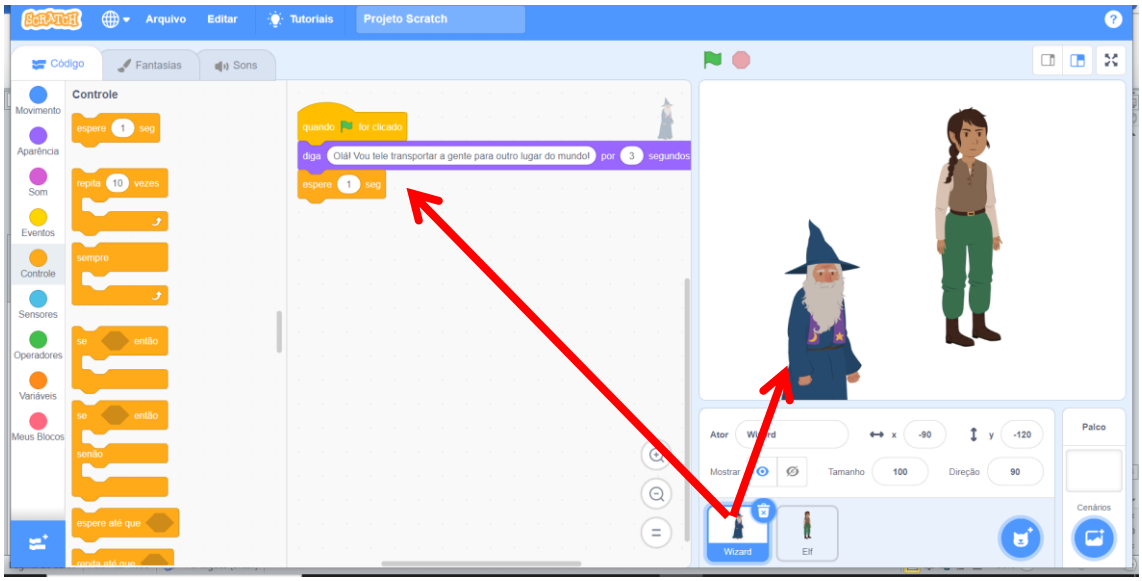
Clique em *imaginários* e o escolha *Elf*



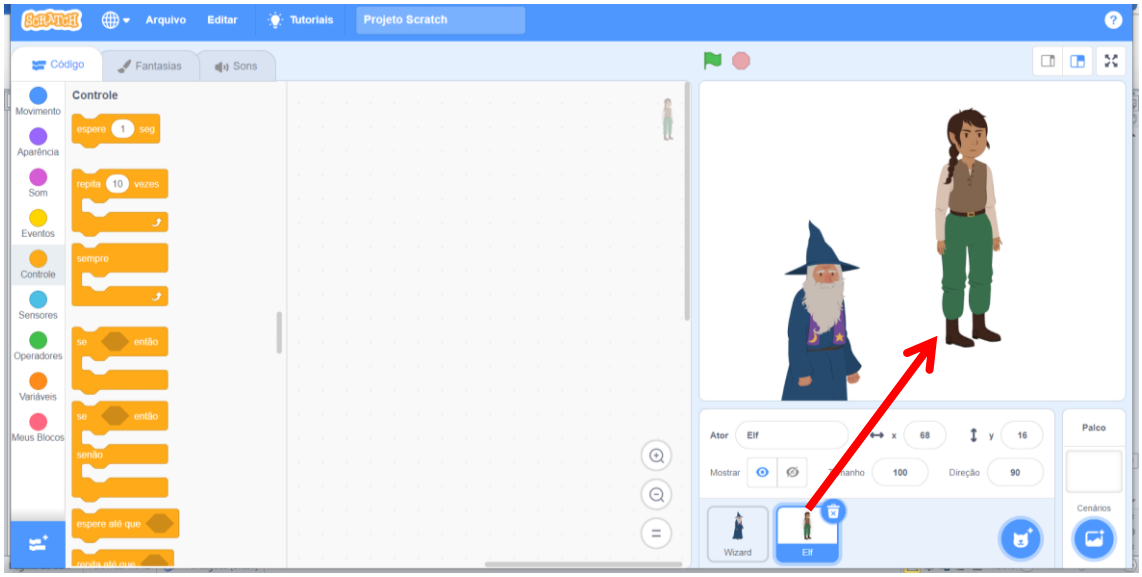
Sua tela deve ficar assim:



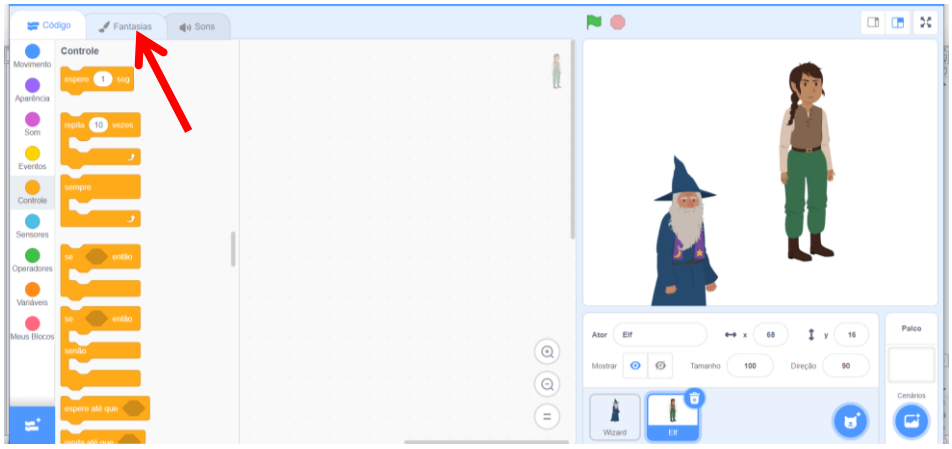
Para ver a programação que fizemos para o mago (wizard), devemos clicar no ato*r wizard.*



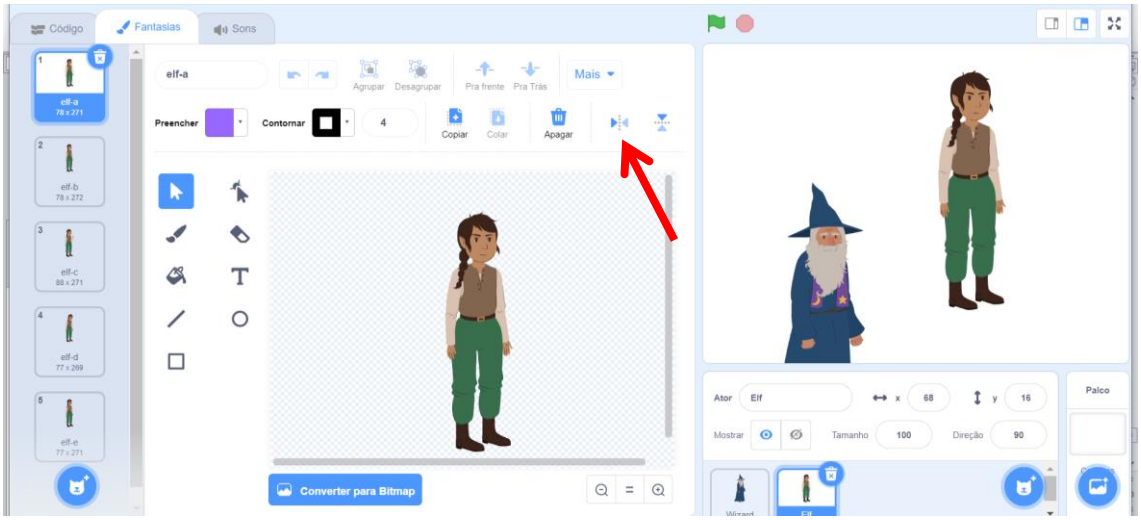
Vamos voltar para o segundo ator.



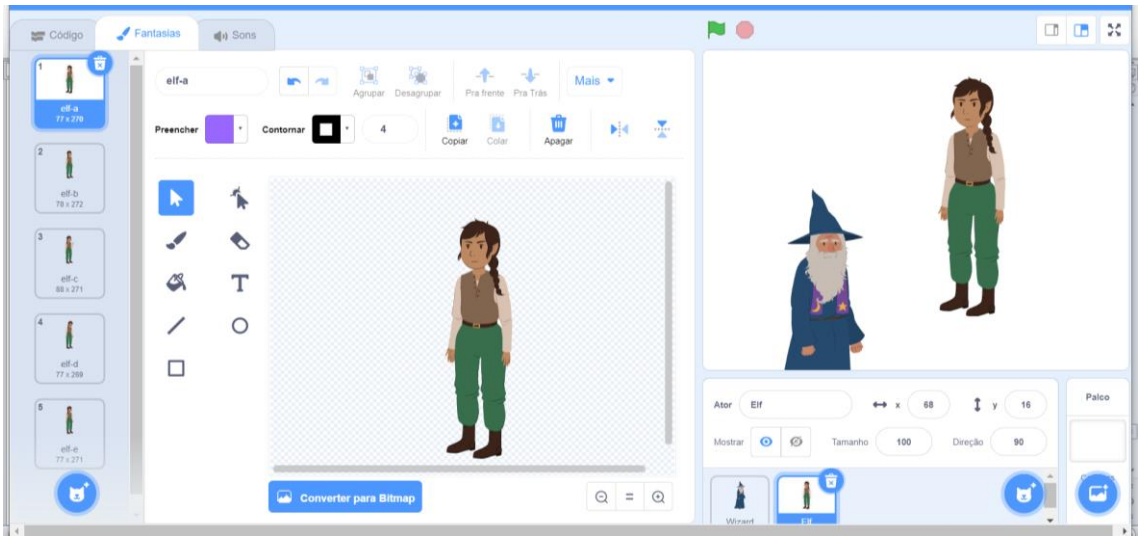
Vamos girar o corpo do ator para que ele fique de frente para o mago (wizard).Clique *fantasia:*



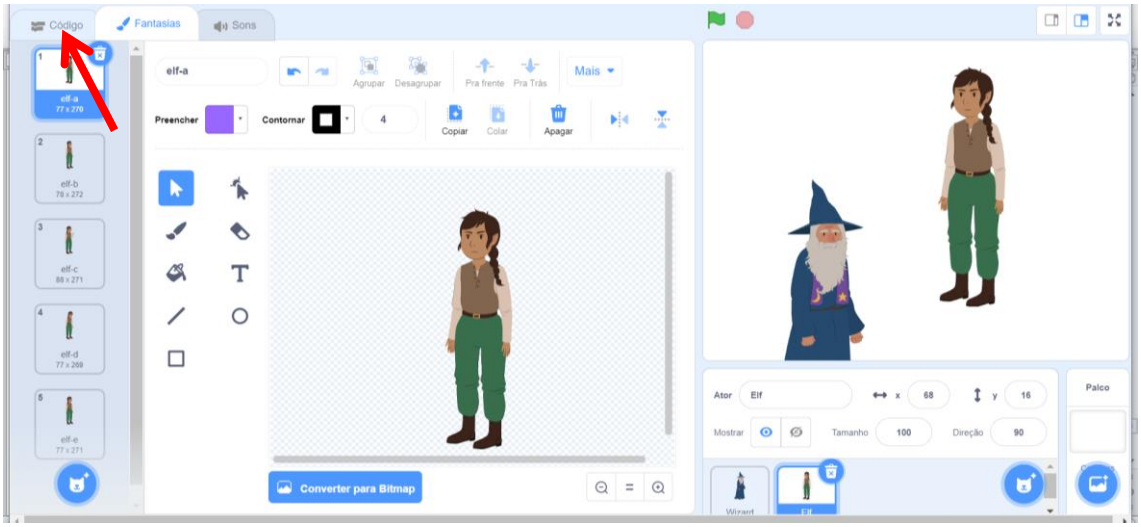
Nessa aba temos todas as fantasias que representam o movimento do ator. Na primeira fantasia, já selecionada, vamos dar um clique em .



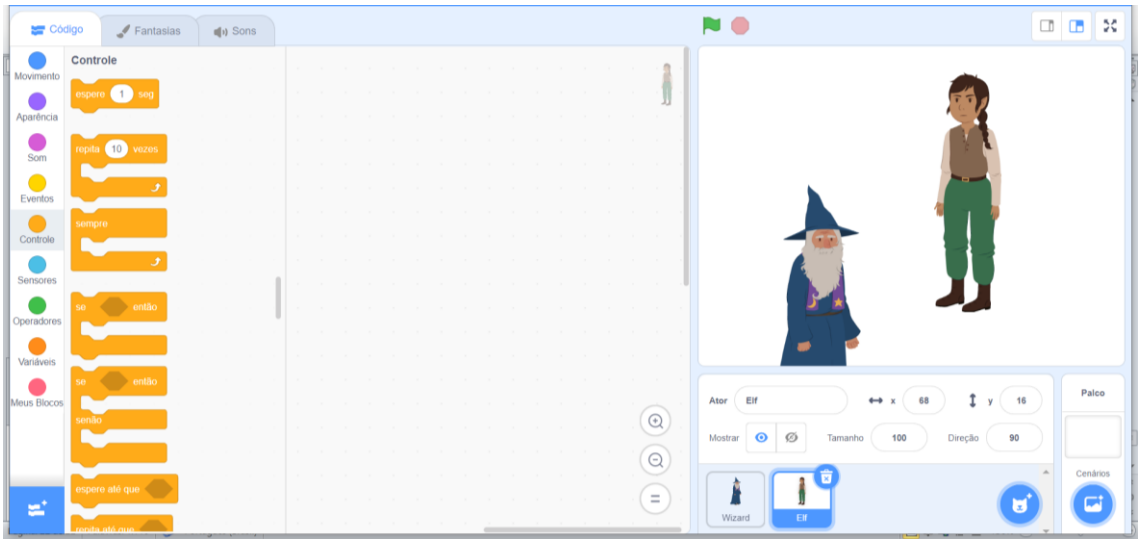
Após o clique o ator vai girar e ficar de frente para o mago.



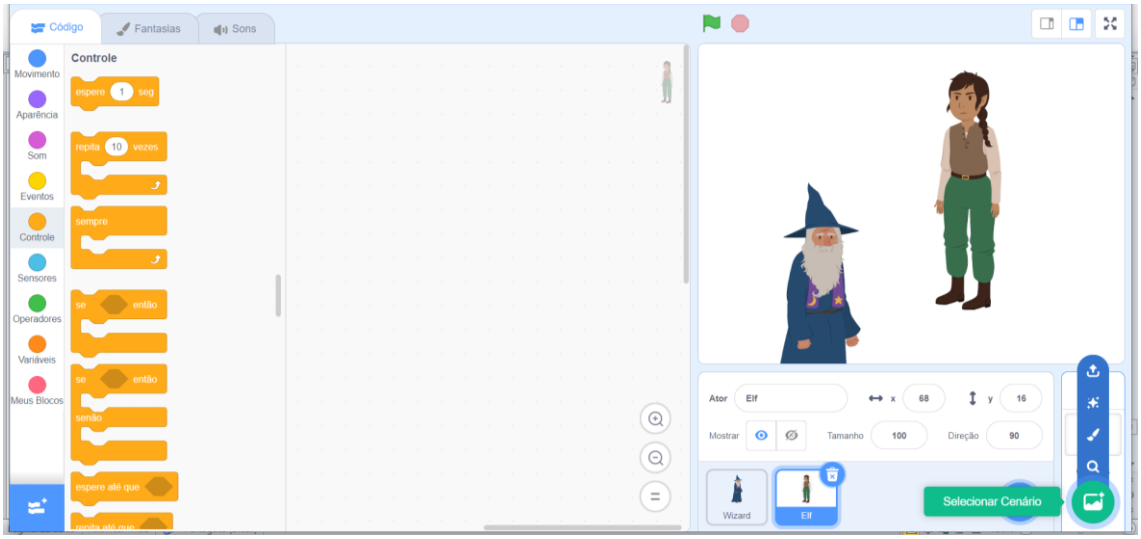
Agora clique em *código* para retornar a tela de programação:



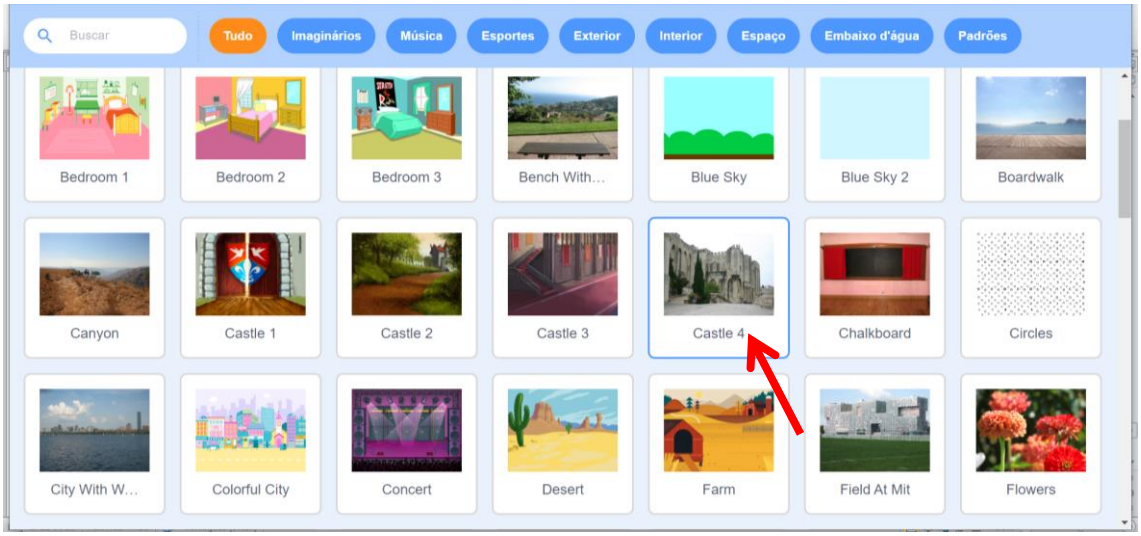
Sua tela deve ficar assim:



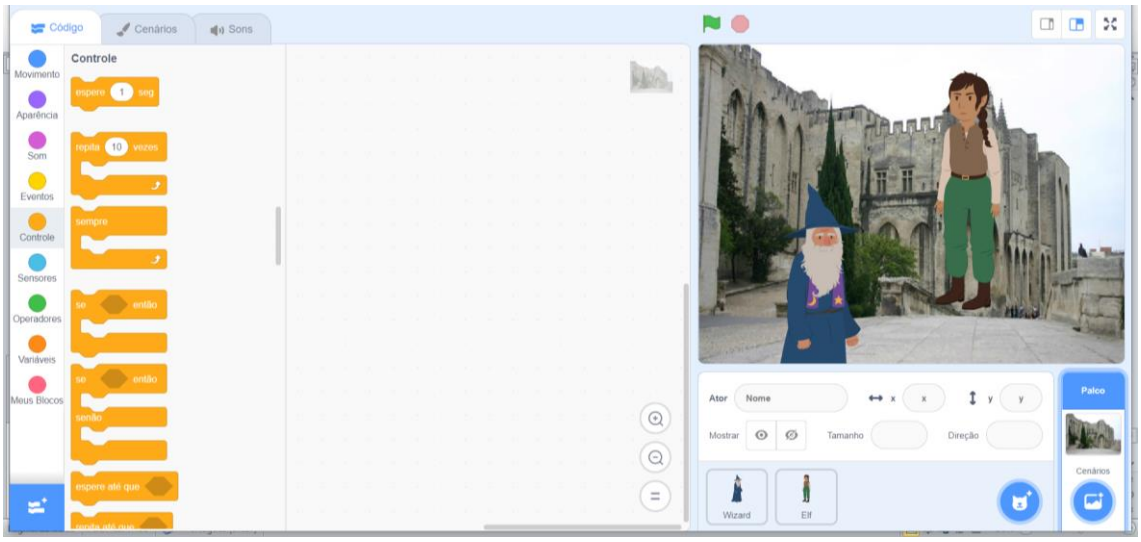
Agora vamos selecionar dois cenários e programar eles. Clique em *Selecionar Cenário:*



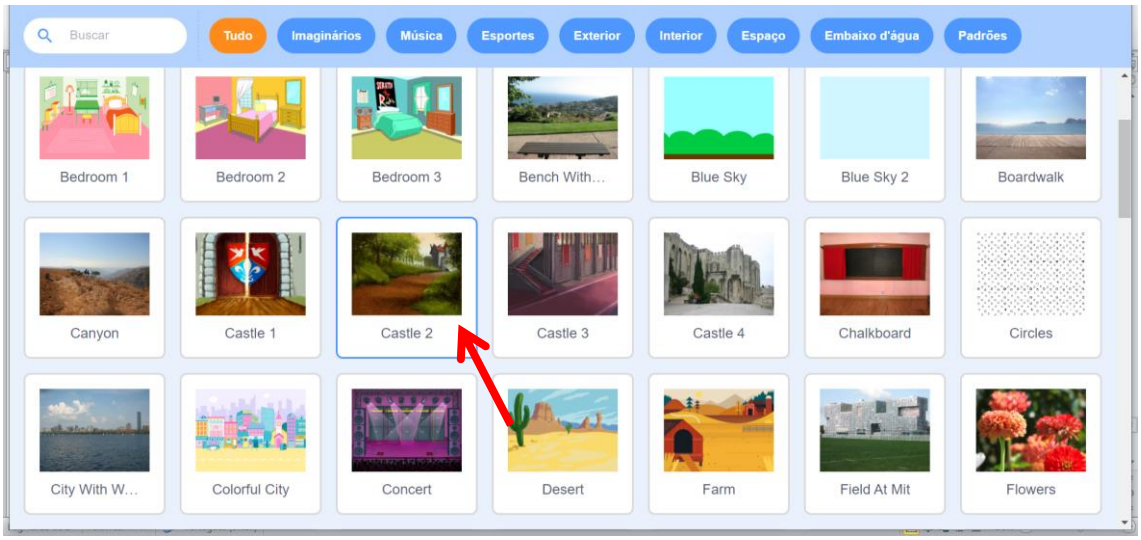
Vamos selecionar o cenário *Castle 4*:



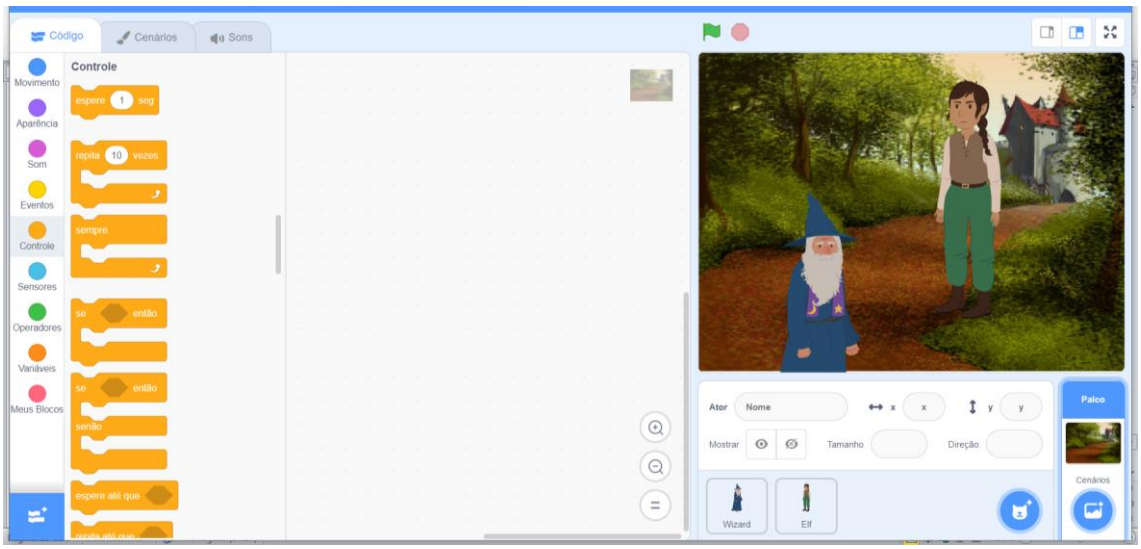
Sua tela deve ficar assim:



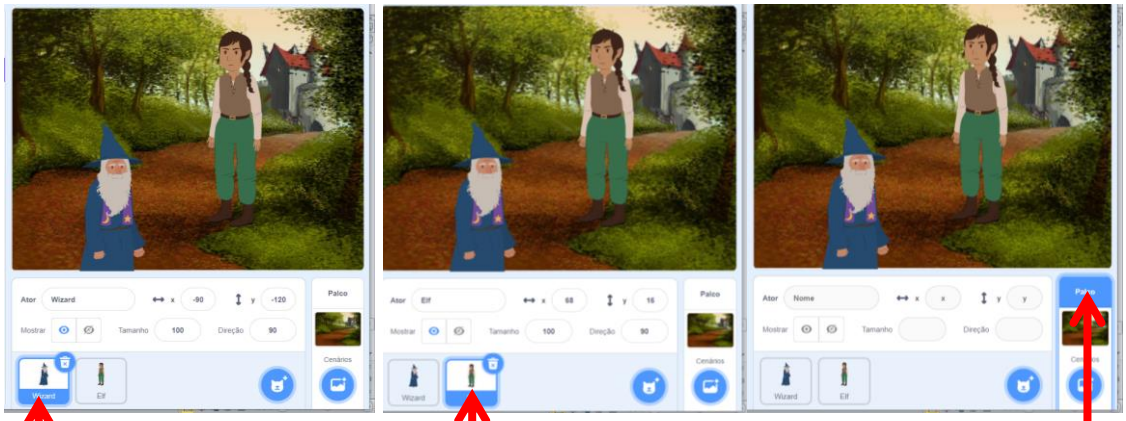
Agora vamos repetir o procedimento em *selecionar cenário* e escolher o segundo cenário. Selecione *Castle 2*



Sua tela deve ficar assim:



Aqui cabe uma observação importante no nosso curso. Estamos trabalhando com dois atores e dois cenários. Cada vez que programarmos ou alterarmos quaisquer um deles, devemos selecioná-lo na barra inferior.

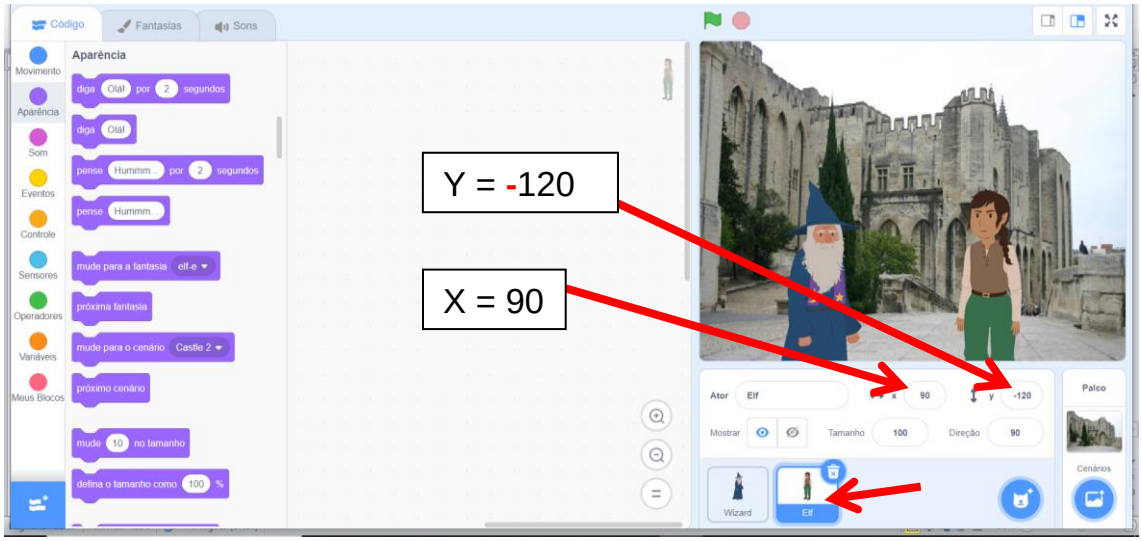


Para trabalharmos com o ator 1 (mago), devemos selecioná-lo.

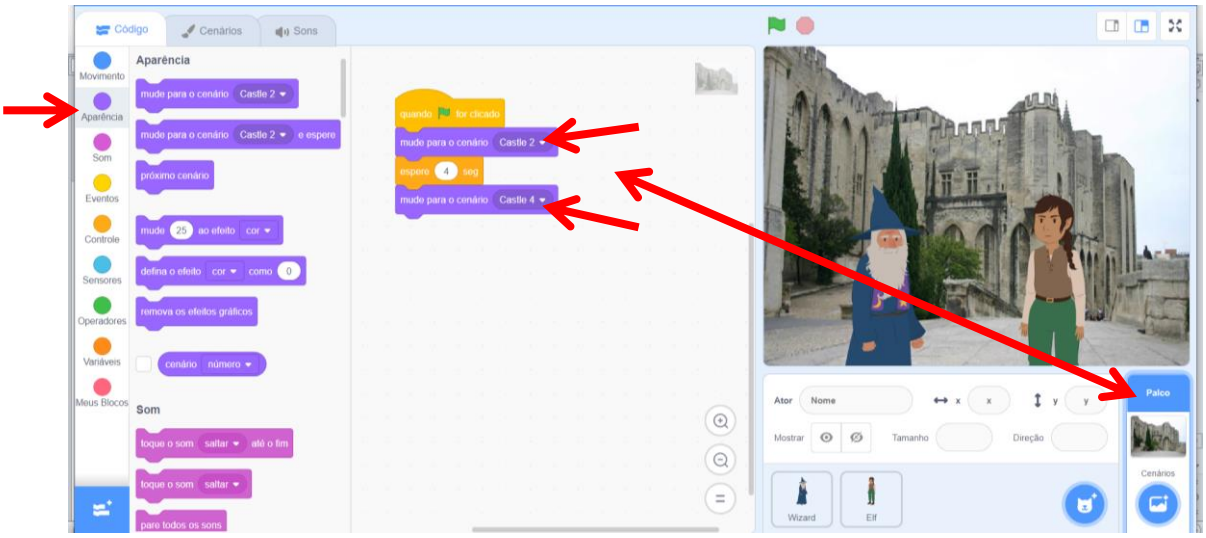
Para trabalharmos com o ator 2 (elfo), devemos selecioná-lo.

Para trabalharmos com os cenários, devemos selecioná-lo em palco.

PASSO 4: Agora vamos selecionar o ator 2 (Elf) e ajustar sua posição na tela com as coordenadas x = 90 e y = -120:



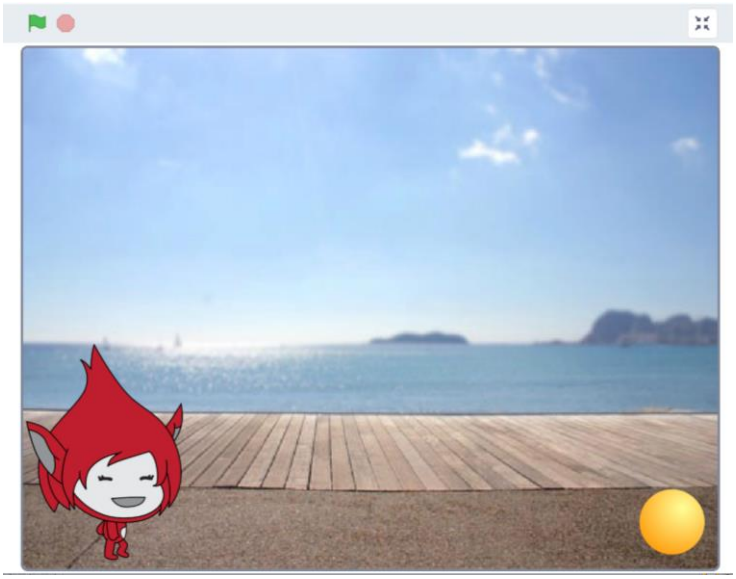
PASSO 5: Vamos programar os cenários agora. Lembre-se que no ator 1(mago), programamos 4 segundos de ação, 3 segundos de fala e 1 segundo de espera, logo, para a troca de cenários, vamos programar um tempo de 4 segundos.



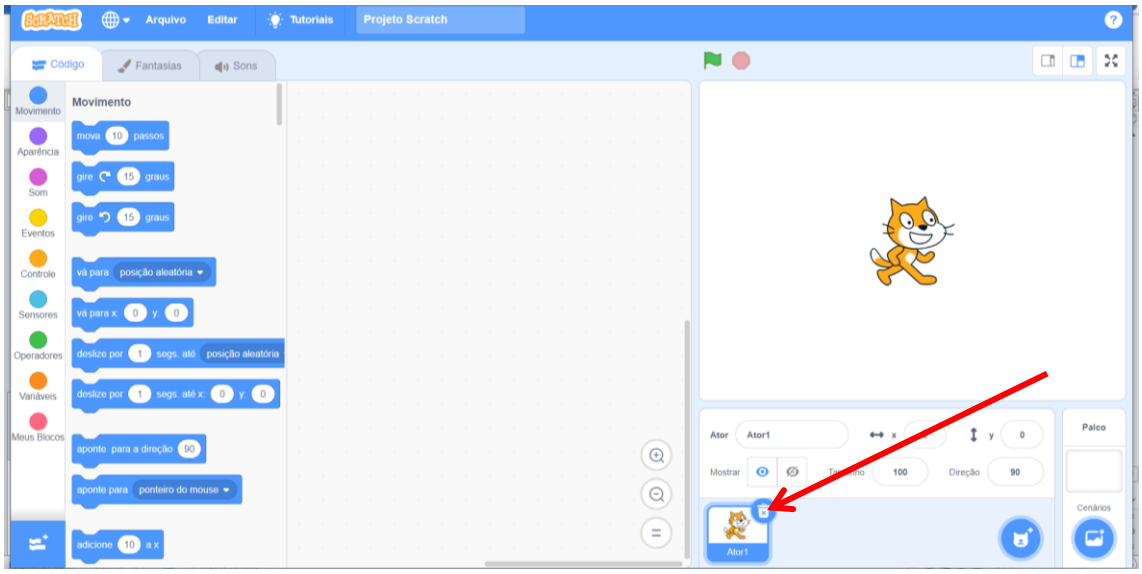
Pronto, chegou a hora de testarmos se nossa programação deu certo. Clique em .

Aula 4: Agora vamos criar um jogo simples. Um bola (ator 1) ficará passando horizontalmente pela tela e nosso segundo ator irá saltar. Para controlá-lo usaremos o teclado.

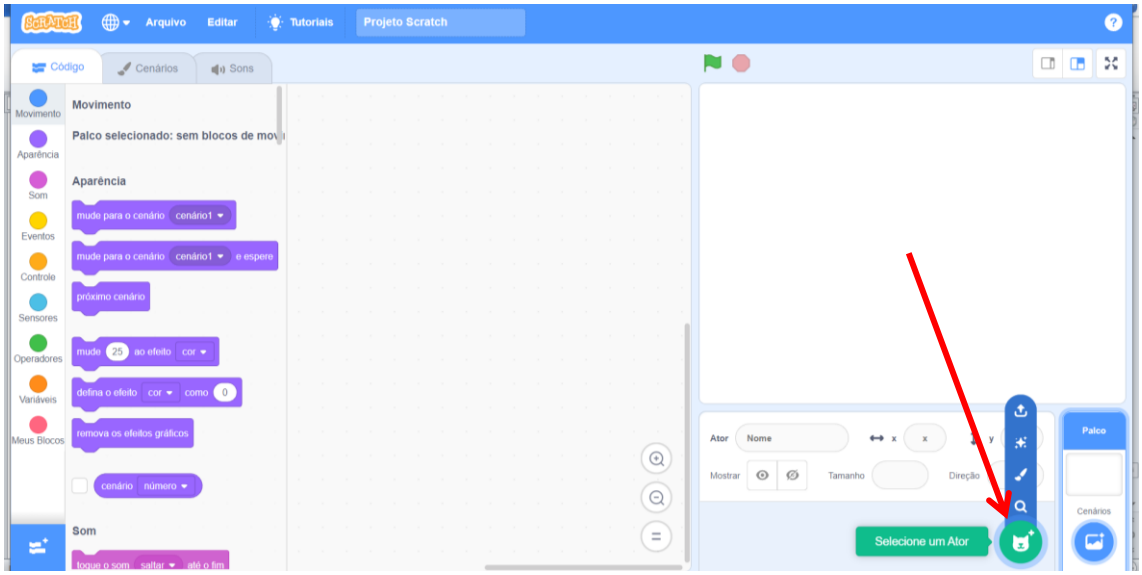
Precisaremos de dois atores: ball (ator 1) e Giga Walking (ator 2), e um cenário: Board Walk.



Passo 1: Abra o Scratch e exclua o ator 1.

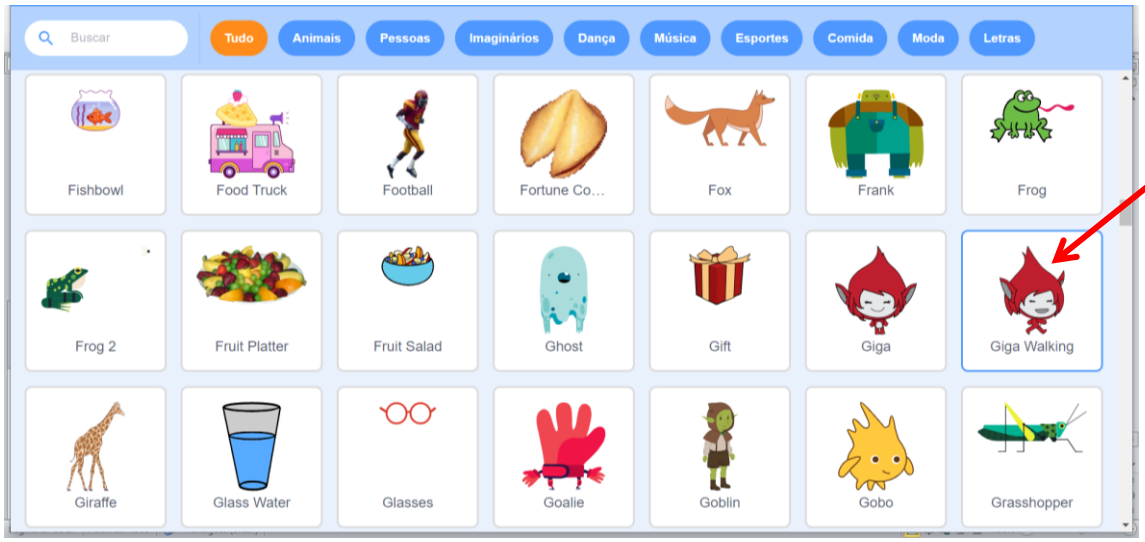


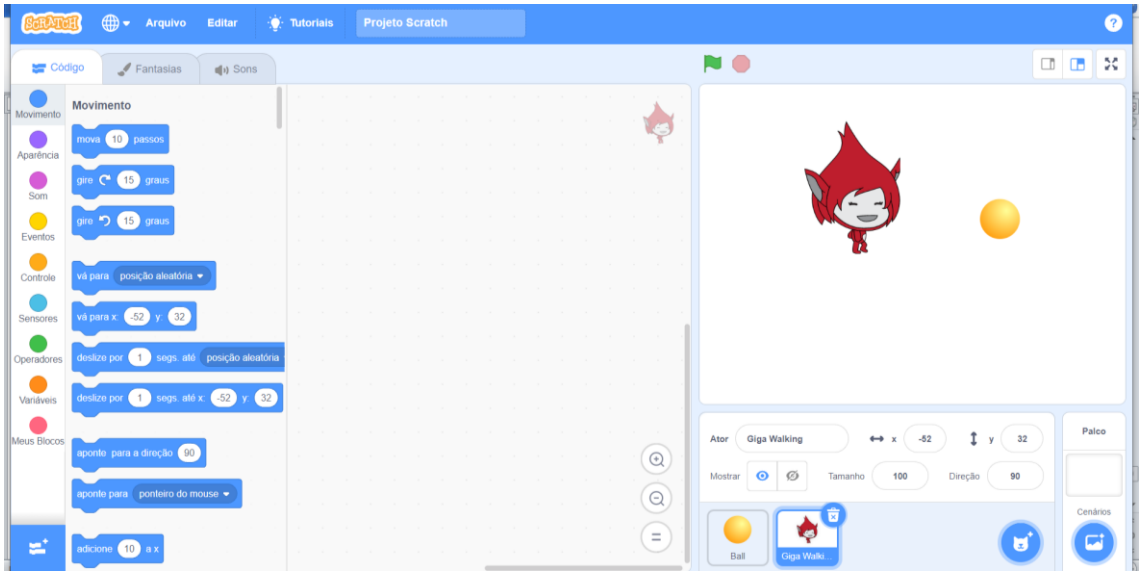
Agora clique em “selecionar um ator”:



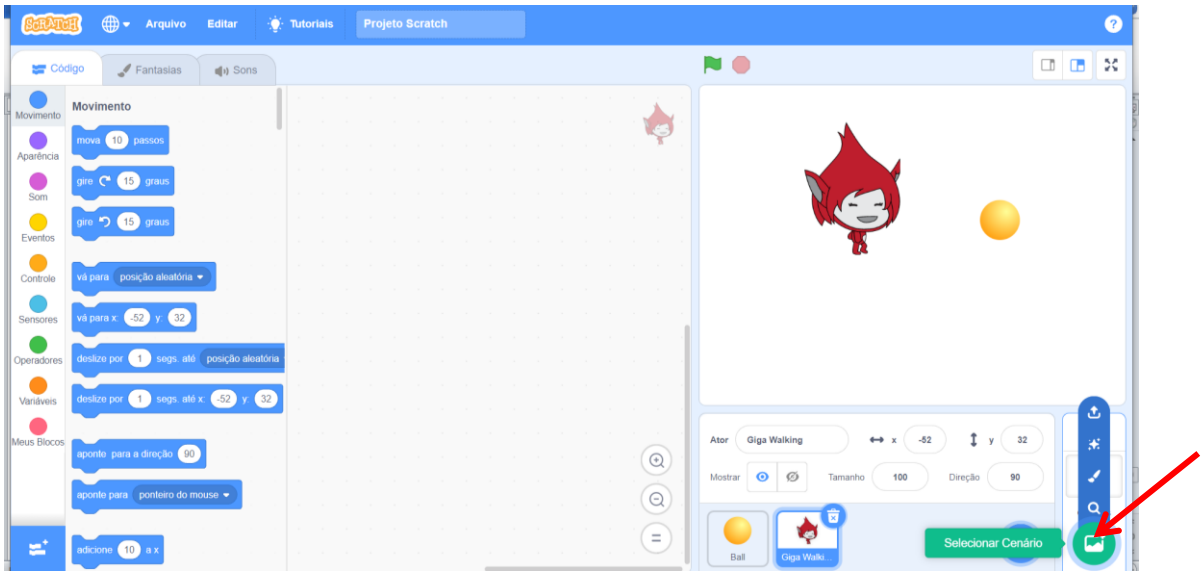
Escolha o personagem “Ball”:

Clique novamente em “Selecionar um ator” e selecione o ator Giga walking.

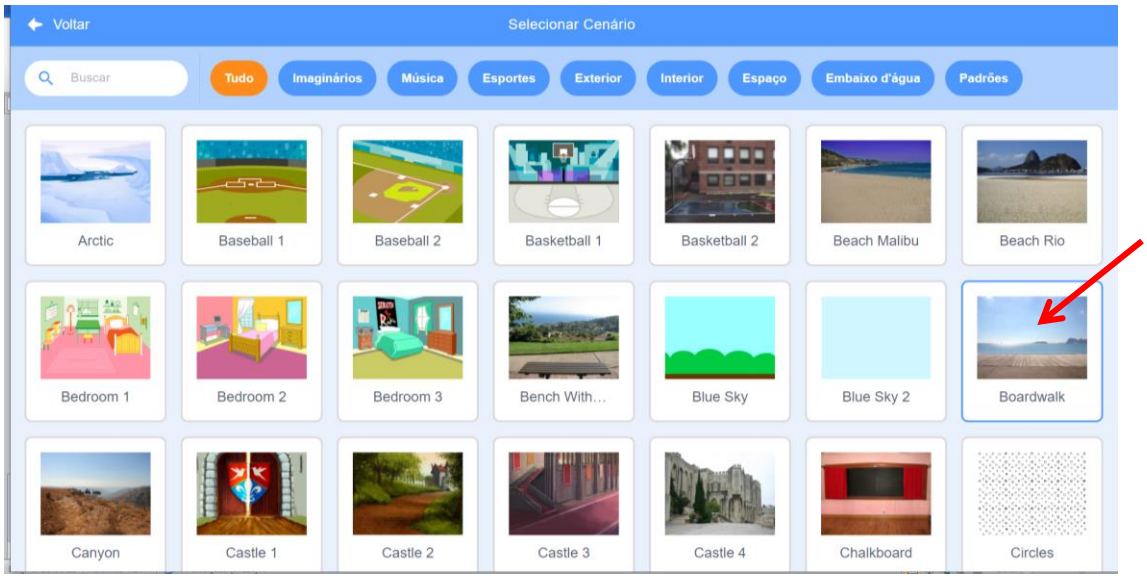




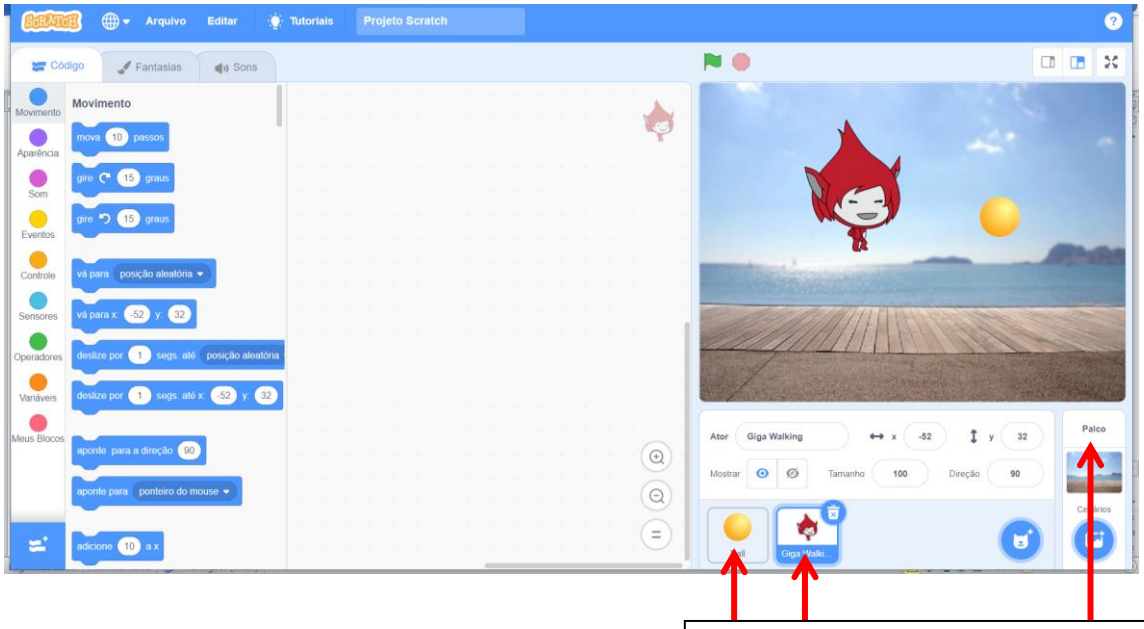
Clique em “Selecionar cenários”:



Selecione o cenário “Boardwalk”:

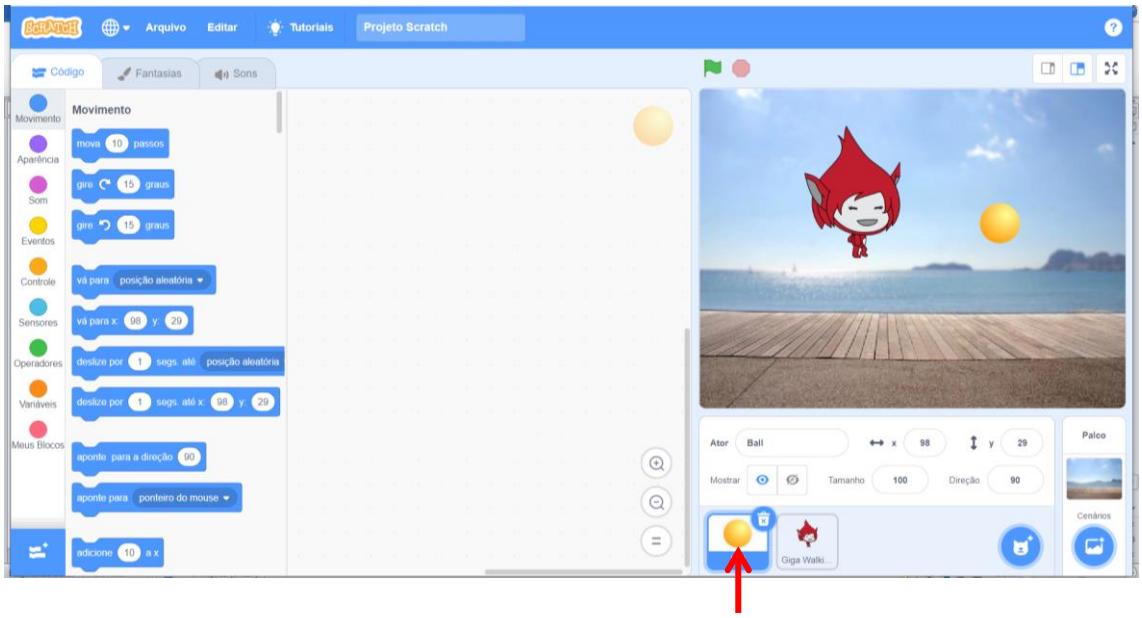


Sua tela deve ficar assim:

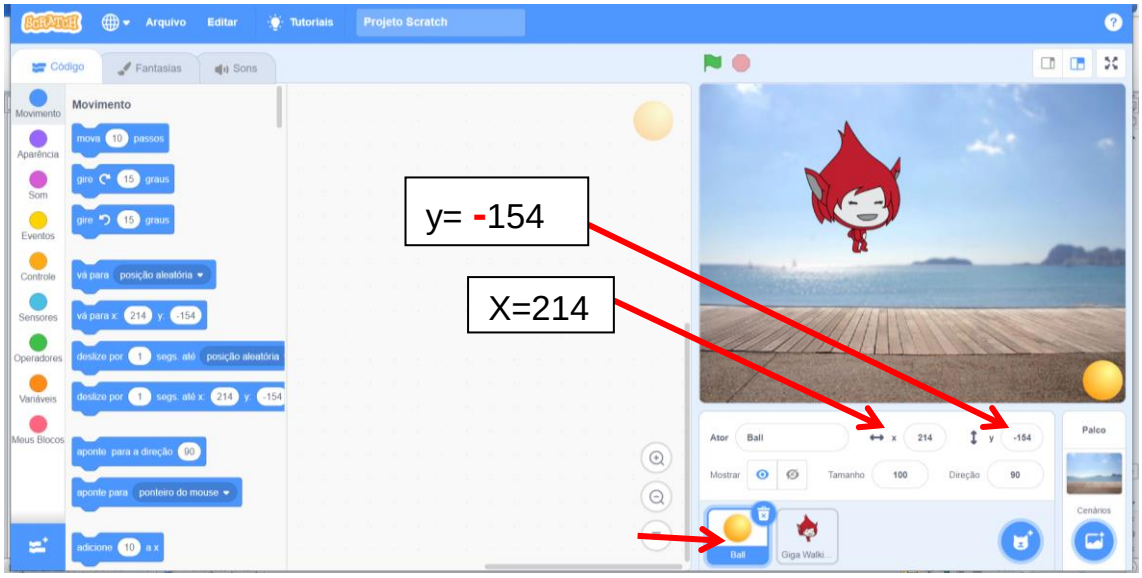


Lembre-se: para programar precisamos selecionar qual ator ou cenário queremos modificar.

Passo 2: primeiro vamos programar o ator 1 (ball), para isso, vamos selecionar esse ator na barra inferior.



Agora podemos começar a programação. Vamos posicionar o ator no canto inferior direito da tela. Para isso vamos deixá-lo com as coordenadas x = 214 e y = -154.



A ideia na **programação** é que o ator ball deslize até a posição x = -230 e y = - 154. Observe que só vamos alterar o valor do x, pois só queremos que ele deslize na horizontal.

Vamos usar essa sequência de comandos:

Comando que inicia a animação.

Observe que essas são as coordenadas iniciais.

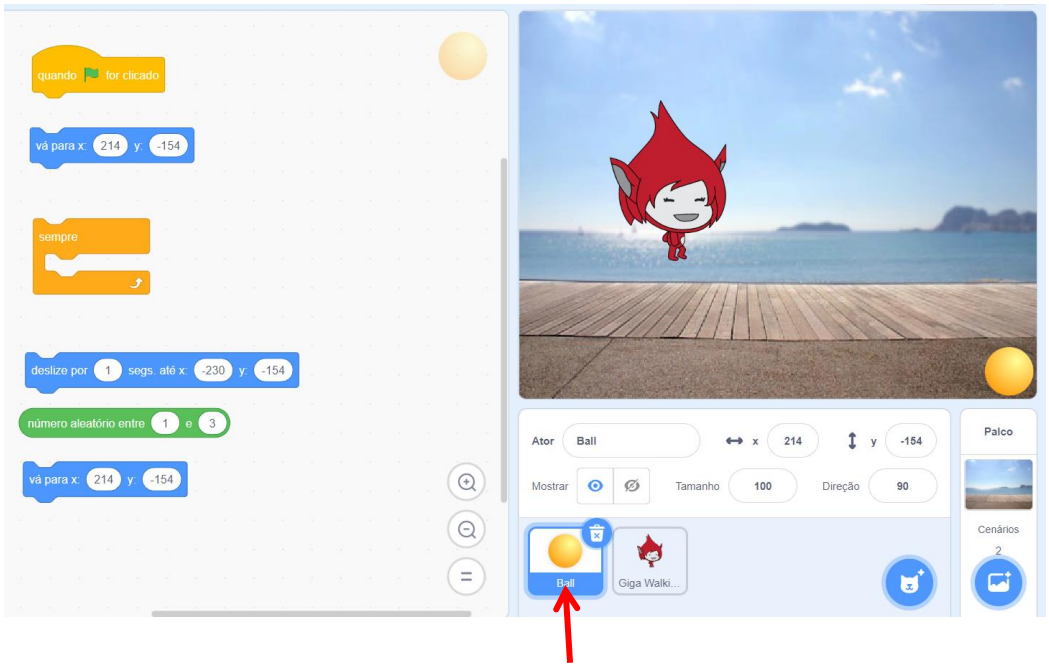
Comando que repete *sempre* os blocos que estão no *seu interior*.

Observe que a coordenada X= - 230 indica o movimento horizontal até o canto inferior esquerdo da tela. Quanto ao tempo iremos alterar. Esse é um comando importante que iremos usar com frequência no nosso curso.

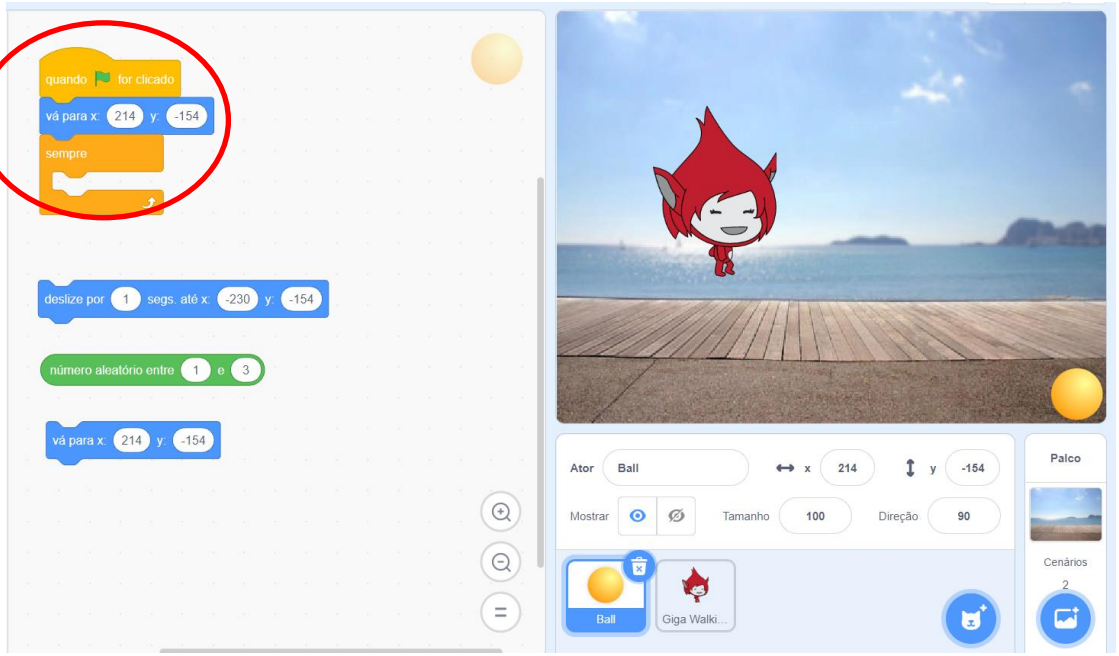
Esse comando escolhe um número aleatório num intervalo que iremos definir.

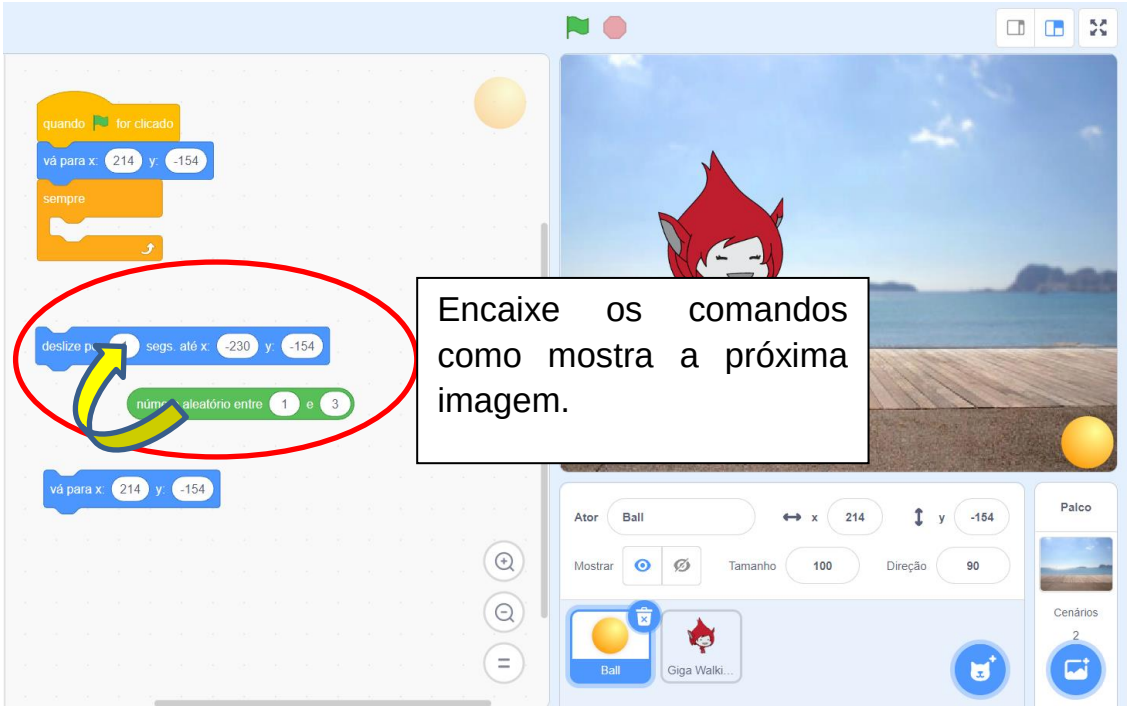
Esse comando vai retornar o ator para sua posição inicial após o movimento.

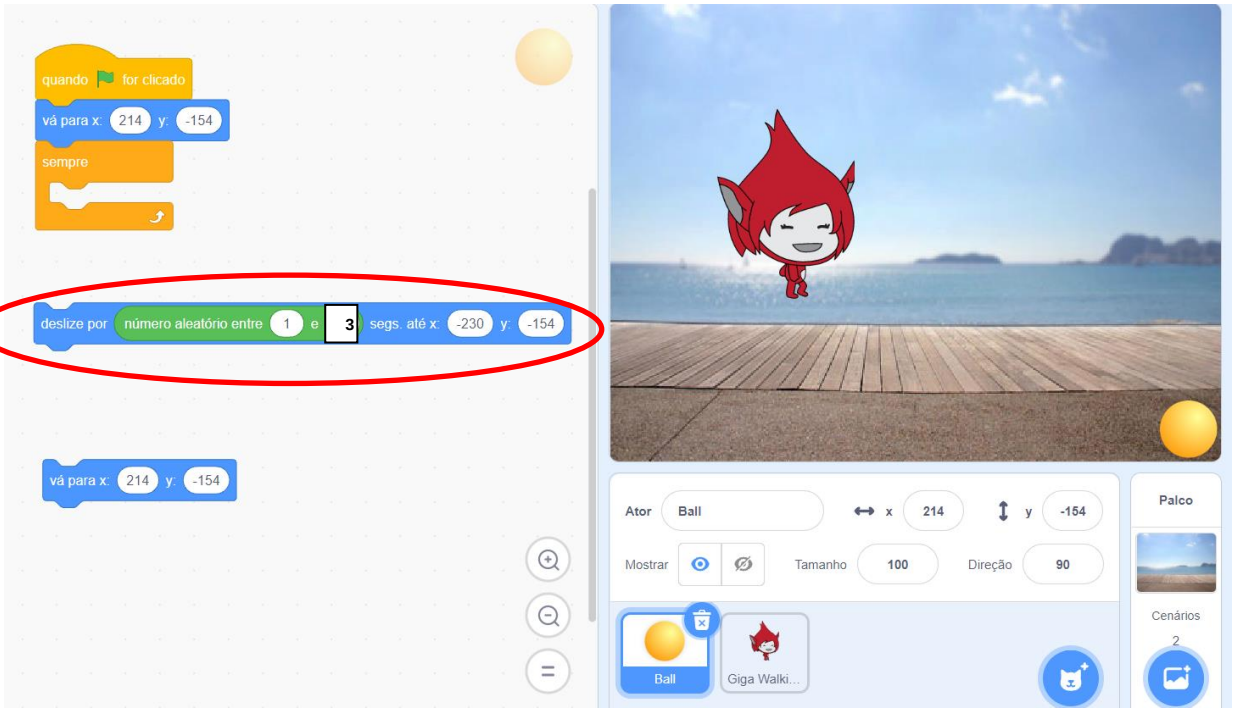
Sua tela, após separar os comandos, deve ficar assim:

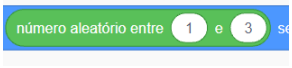


Agora vamos encaixar os blocos na lógica da programação:

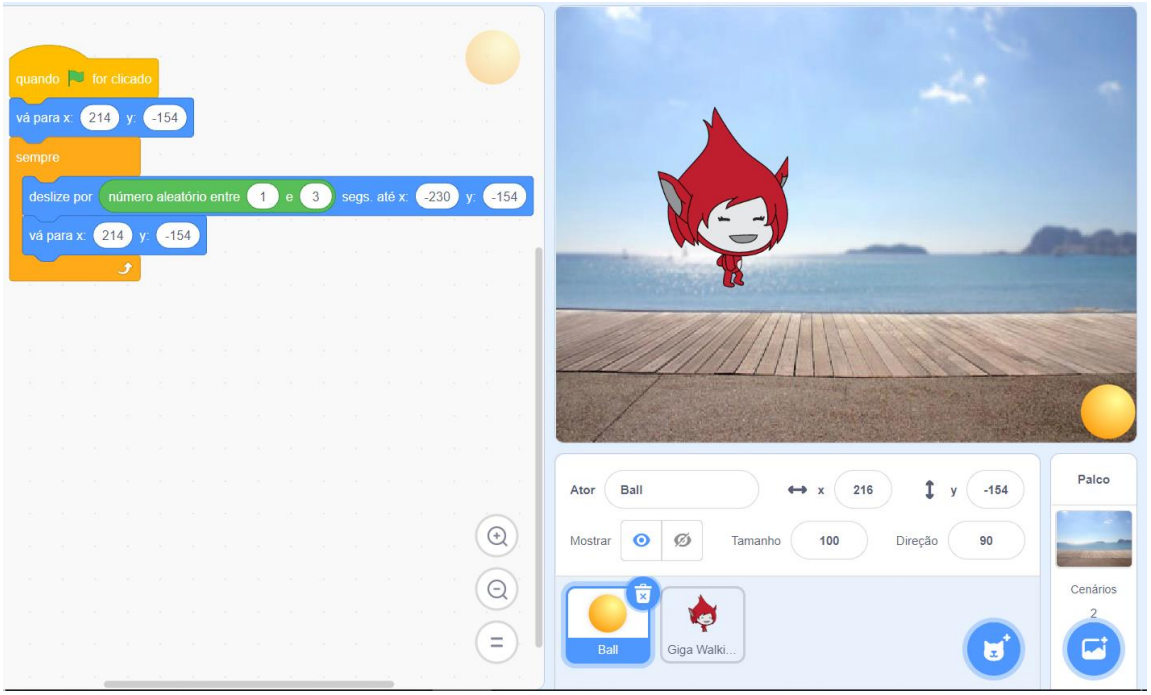




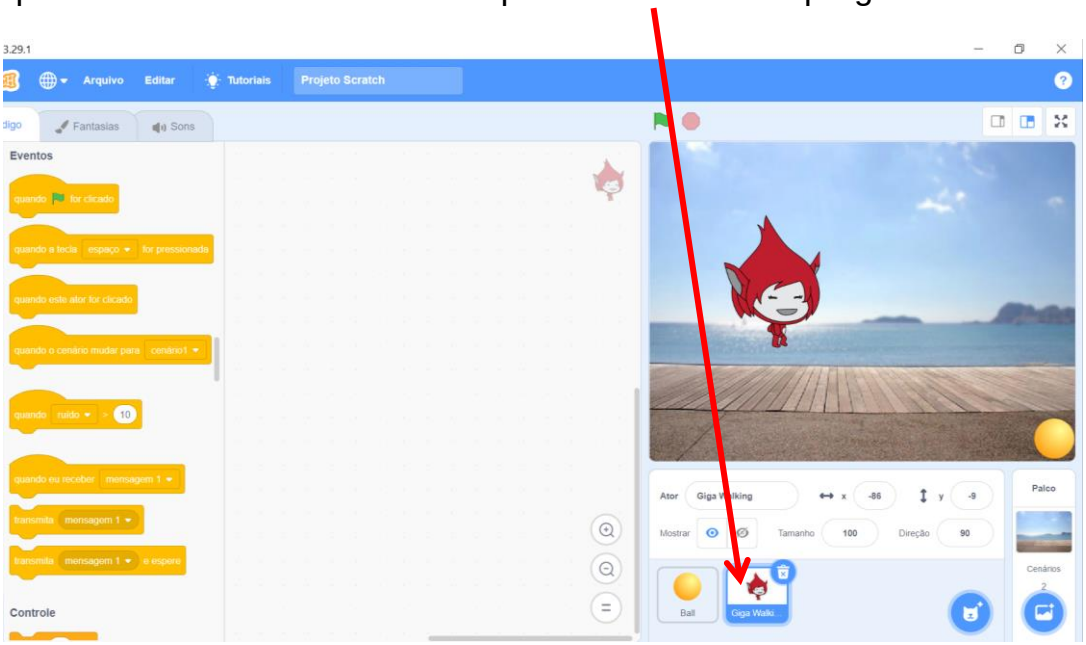


Vamos alterar os de para  e encaixar no comando *sempre.*

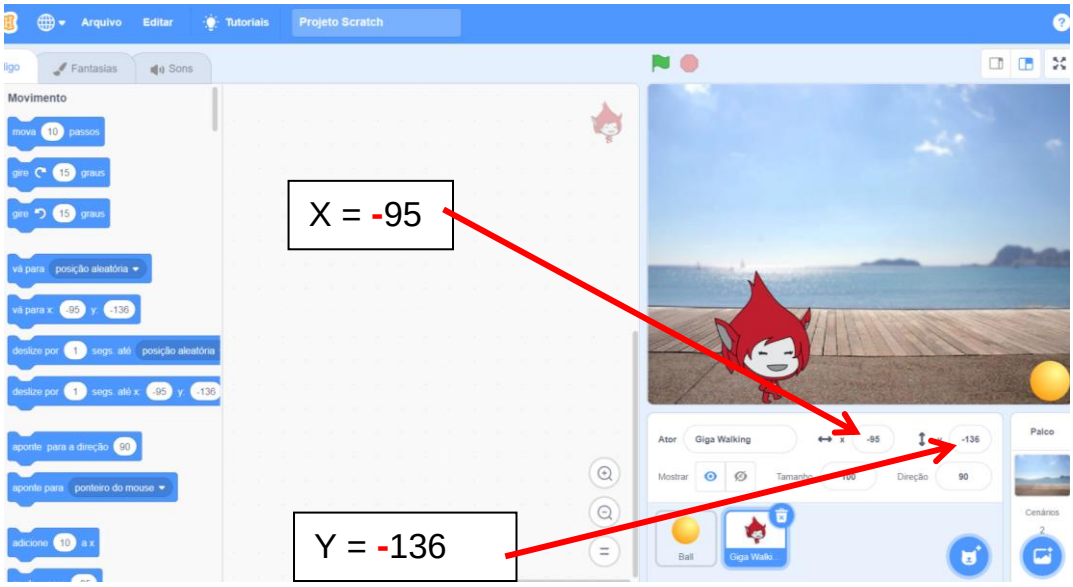
Agora faça um teste e **entenda a lógica da programação**



Passo 2: Agora vamos programar o ator 2 (Giga Walking). Lembre-se que você deve clicar nesse ator para selecioná-lo e programar.



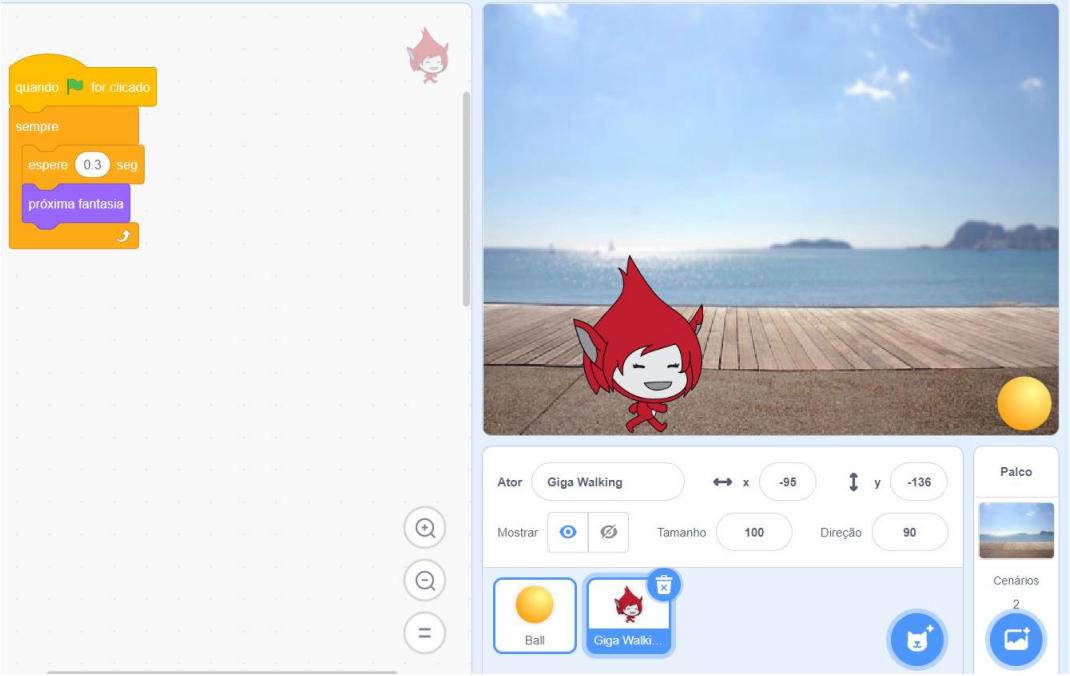
Primeiro vamos posicioná-lo na posição x = -95 e y = -136.



Para programarmos o movimento dando a sensação que o personagem está andando vamos usar o seguinte bloco:

Aqui vamos alterar para 0.3 segundos (*não se esqueça de colocar o ponto antes do 3)

Faça um teste para verificar se o personagem está movimentando as pernas.



Agora vamos criar outro bloco de comando para que nosso personagem salte a bola. Lembre-se que são os valores **do eixo Y** que faz nosso personagem se **deslocar na vertical.**

Iremos usar esses blocos:

Comando **condicional se.....então....**

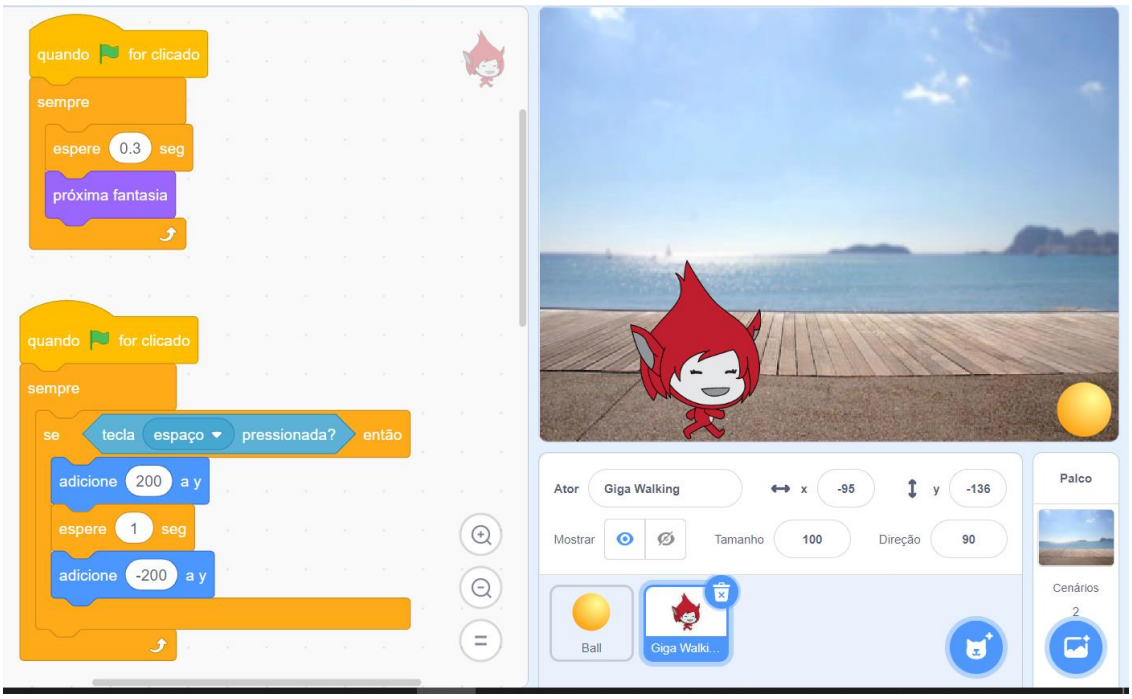
Condição que queremos para acionar os blocos seguintes.

Satisfeita a condição, iremos adicionar **200** unidades no **eixo y**. Isso fará com que o personagem se desloque na vertical (subir)

Espera de 1 segundo antes do próximo comando

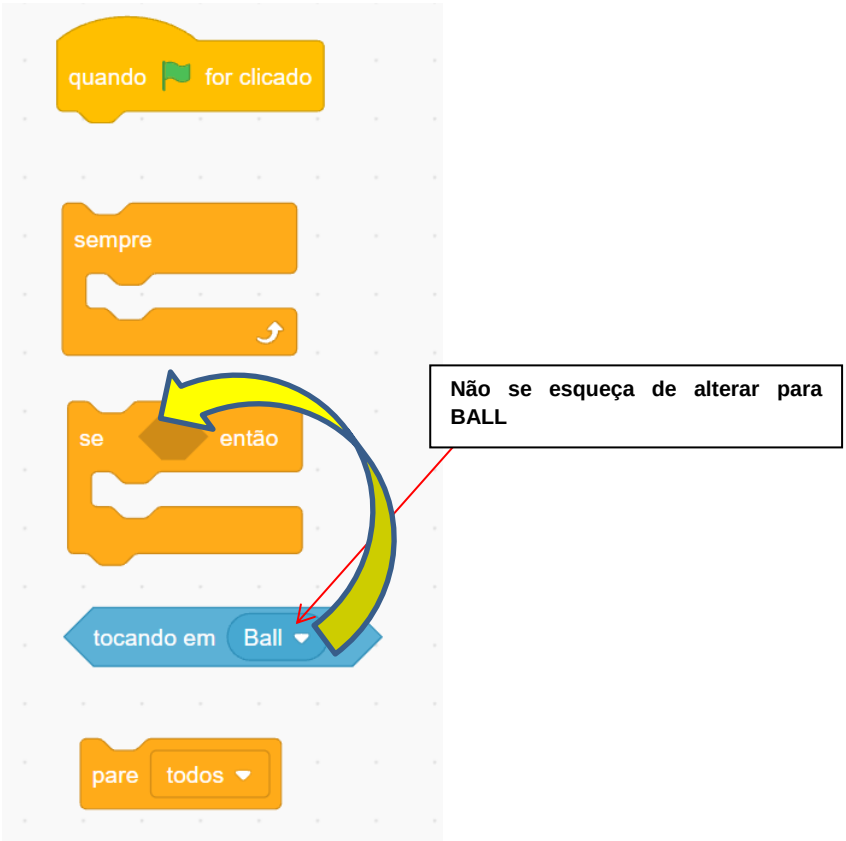
Agora precisamos que ele desça. Para isso, vamos subtrair **200** unidades no **eixo y, ou seja, y = -200**. Isso fará com que o personagem se desloque na vertical (descer)

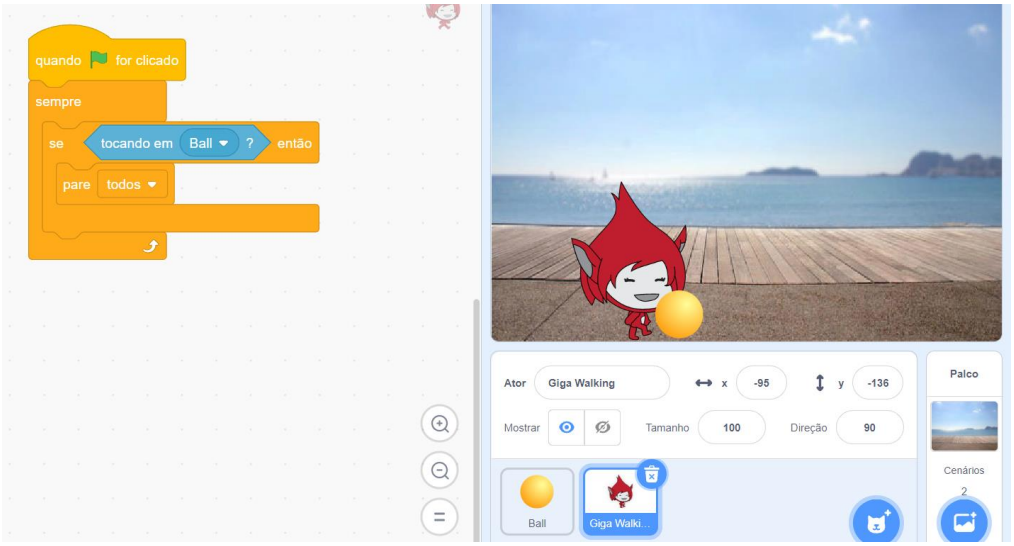
Sua tela deve ficar assim:



Faça um teste e verifique que quando acionado a tecla “espaço” o personagem salta e volta.

O último bloco de comandos do nosso jogo fará a animação parar sempre que o personagem tocar na bola. Vamos usar os comandos:





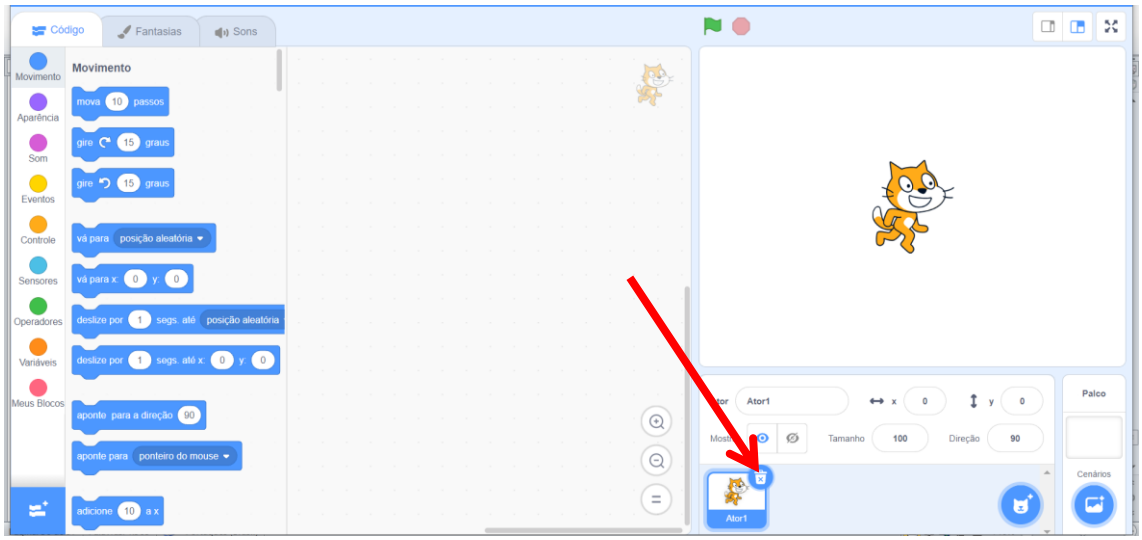
Agora faça o teste do jogo.

Caro (a) estudante, fizemos quatro aulas de conceitos básicos para iniciantes na programação em blocos. Nas próximas aulas iremos aprofundar um pouco mais nos conceitos de lógica de programação focando no desenvolvimento de jogos. Cabe ressaltar que o **objetivo do curso é promover o desenvolvimento cognitivo, o desenvolvimento do raciocínio lógico, a melhora na aprendizagem em Matemática, o lidar com frustações, além do estímulo a imaginação, autonomia e ao trabalho em equipe.**

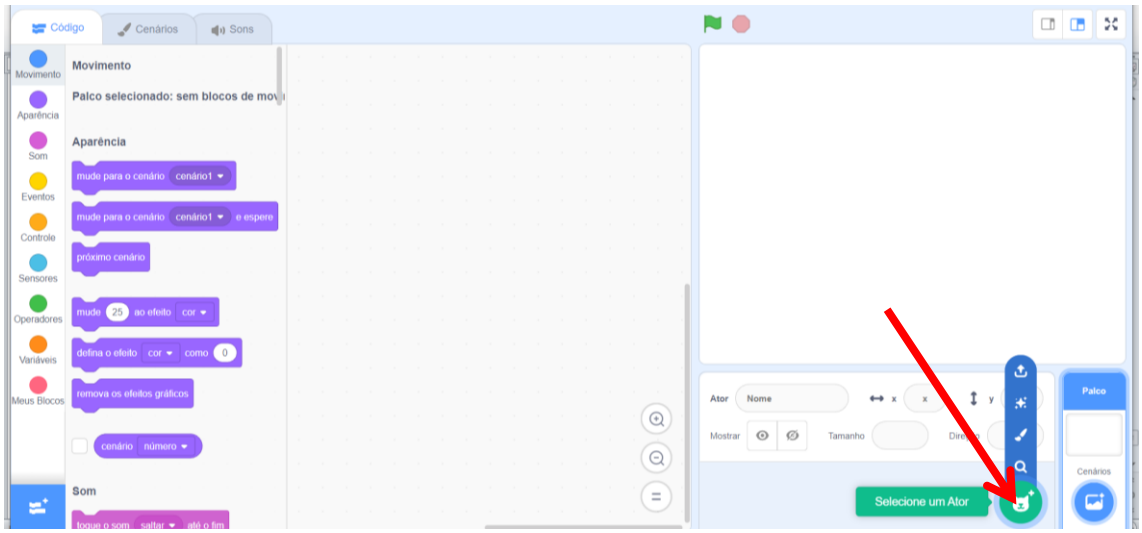
****

Aula 5: Nesta aula vamos criar um jogo em que controlaremos o movimento do personagem com o mouse. O objetivo do jogo é escapar de um inimigo que ficará se movimentando em todas as direções na tela.

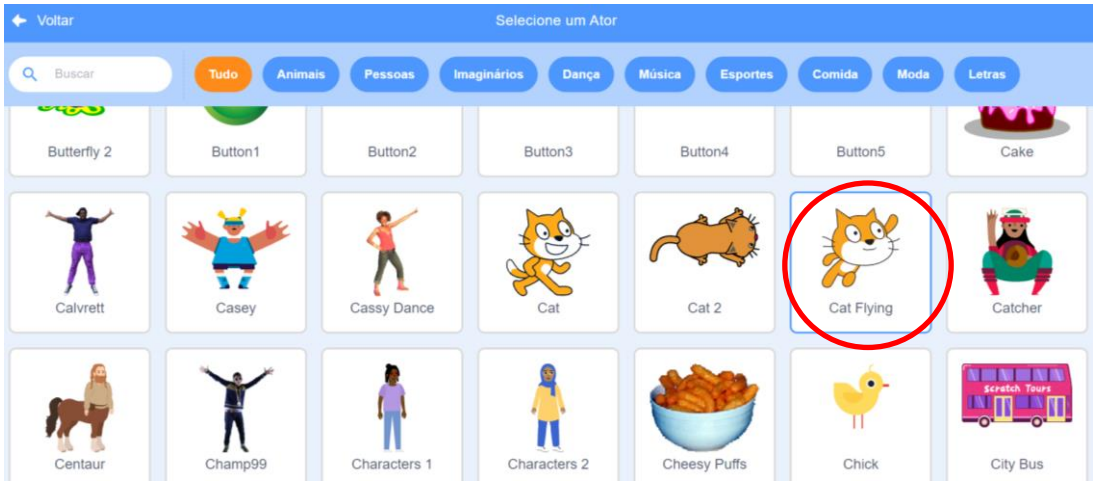
Abra o Scratch e **apague** o personagem *cat*



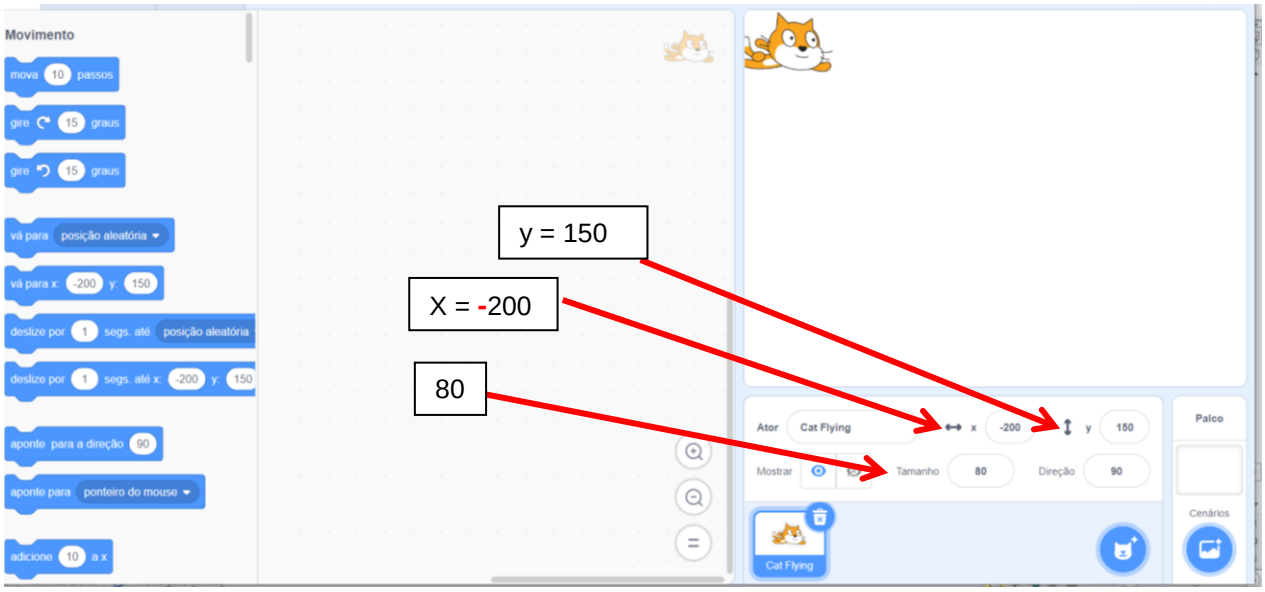
Agora clique em *selecionar um ator*



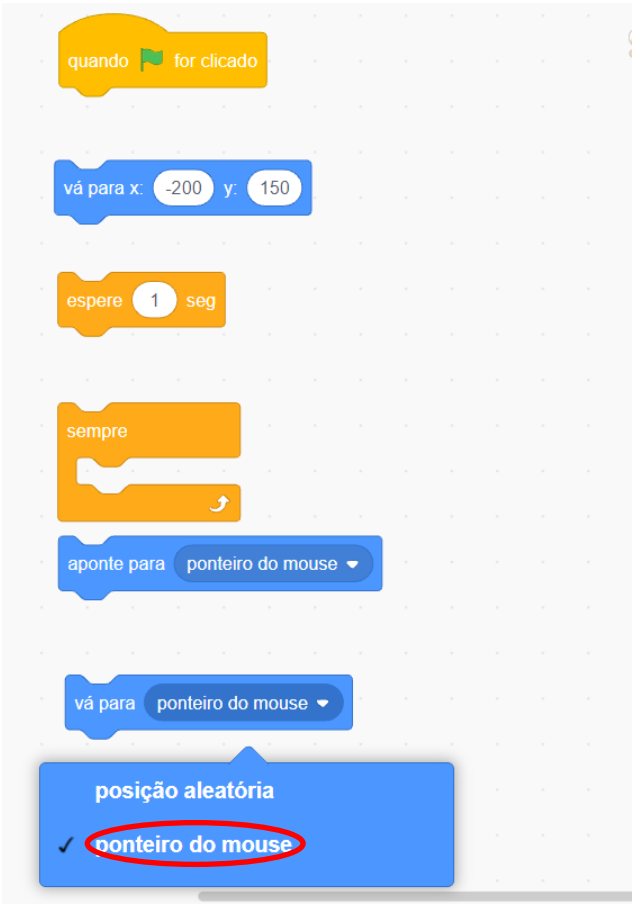
Selecione o personagem CAT FLYING



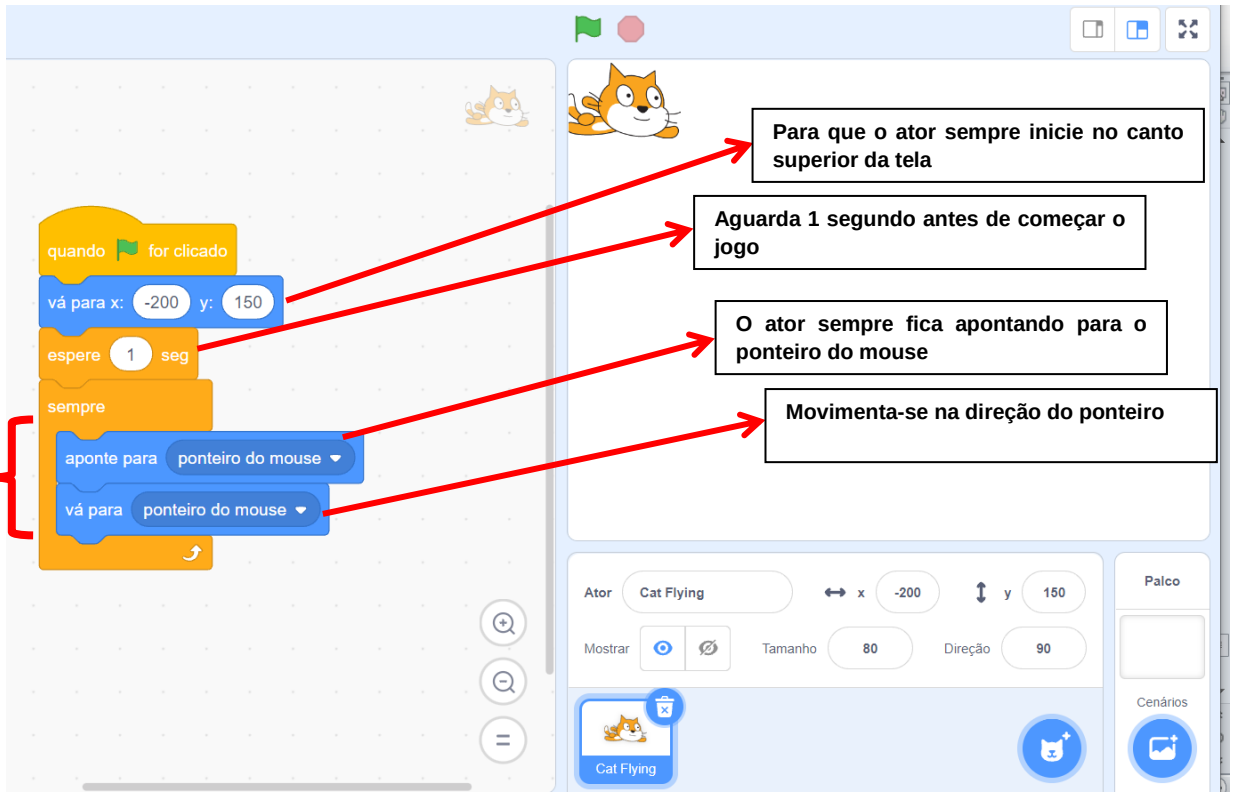
Vamos posicioná-lo na tela no canto superior esquerdo. Para isso, vamos escrever os seguintes valores nas coordenadas: x = -200 e y = 150. E também vamos diminuir o seu tamanho, o reduzindo para 80%.



Agora vamos programar o personagem para que ele se movimente sempre na direção do ponteiro do mouse. Vamos usar os seguintes comandos:

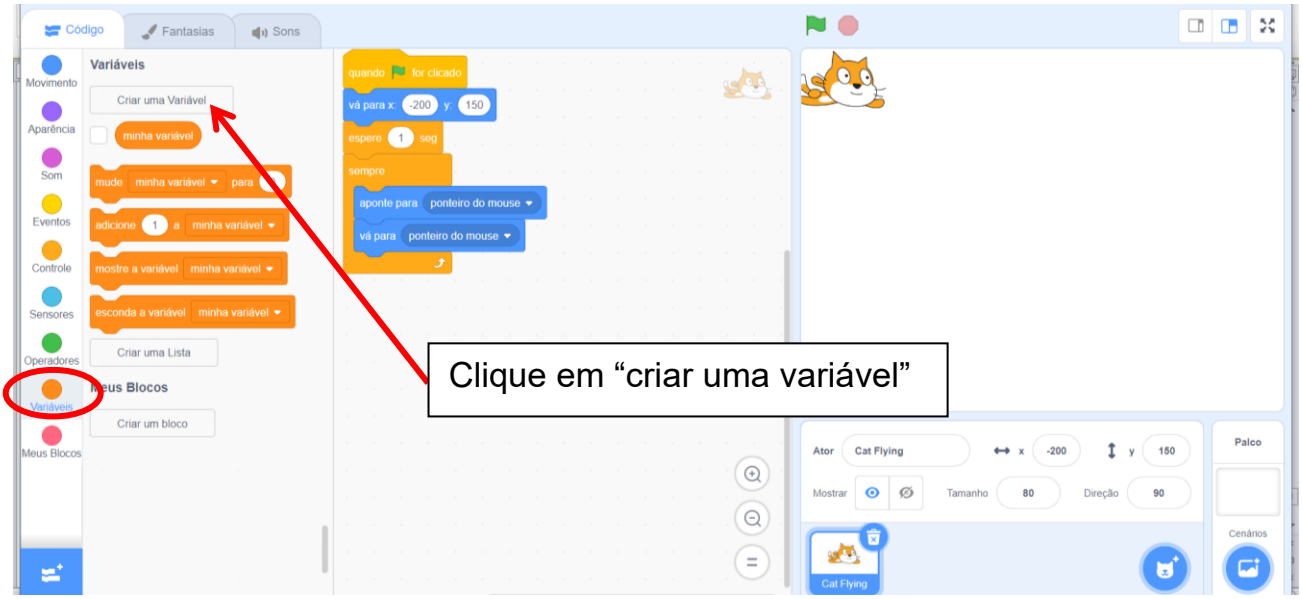


Sua tela deve ficar assim:

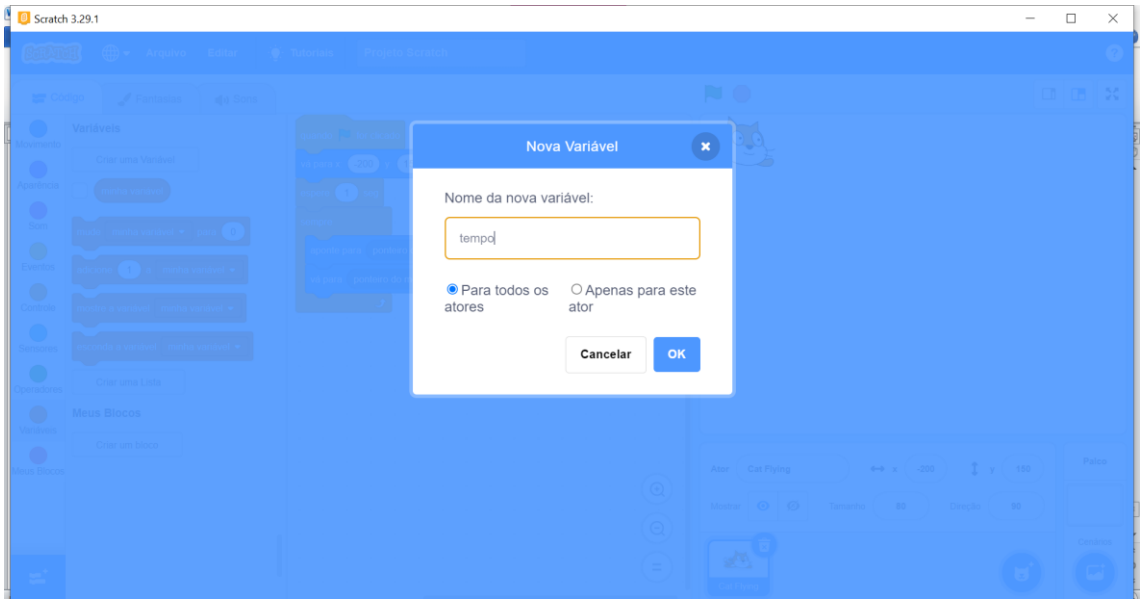


Faça um teste e verifique se o personagem está se movimentando.

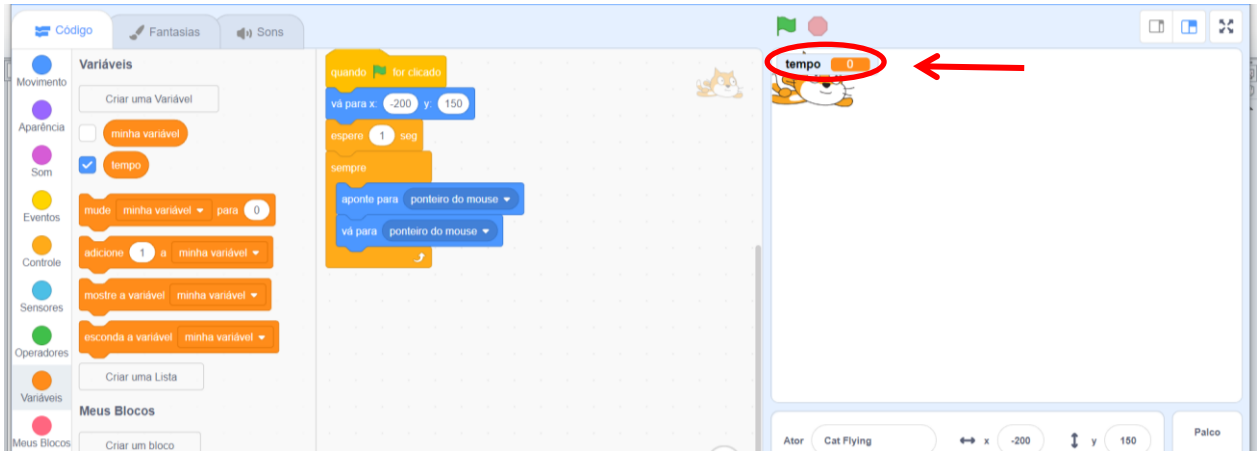
Agora vamos criar um contador de tempo no jogo. Primeiro vamos utilizar um recurso que usaremos com bastante frequência nas próximas aulas: **a criação de uma variável.**



Em “nome da variável” escreva “tempo” e clique em “ok”

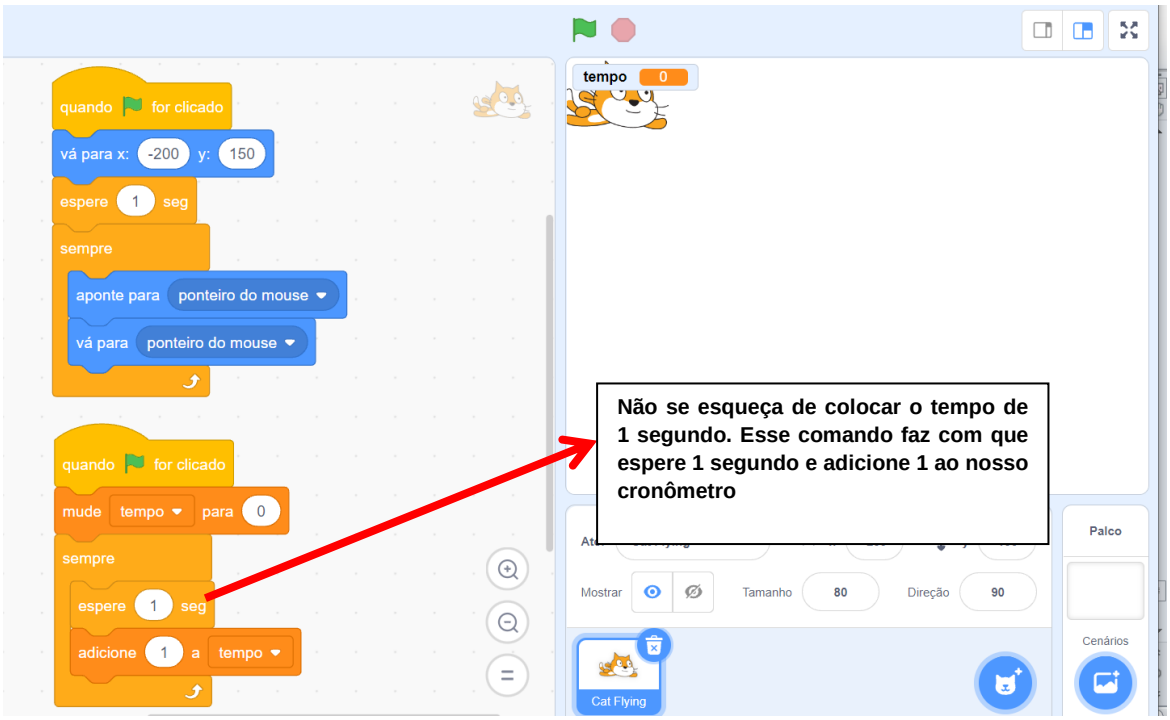


Observe que a variável “tempo” vai aparecer na tela:



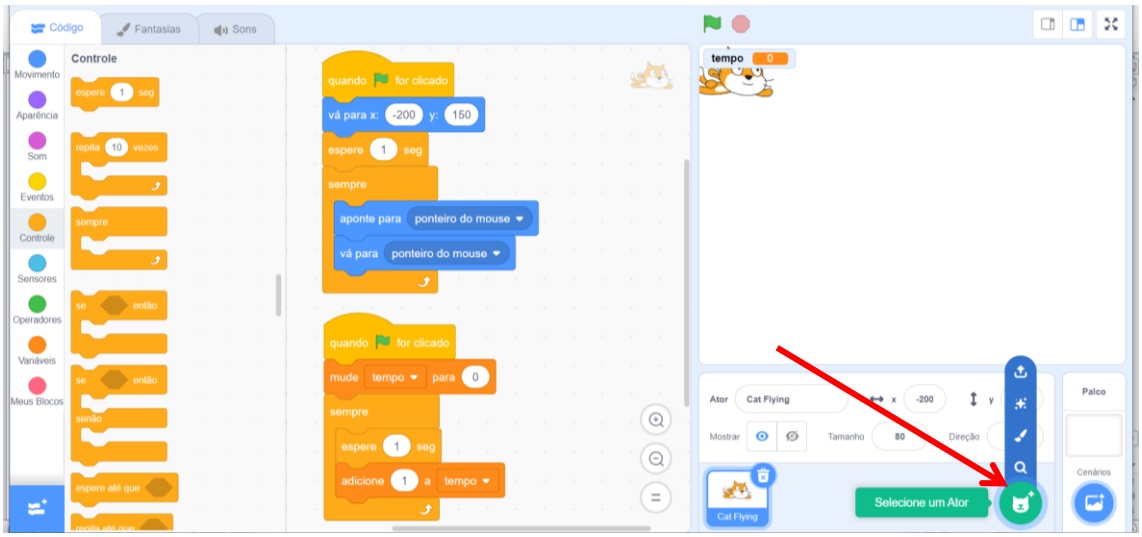
Agora vamos programar para que essa variável inicie a contagem. Para isso, usaremos os seguintes comandos:

Sua tela deve ficar assim:

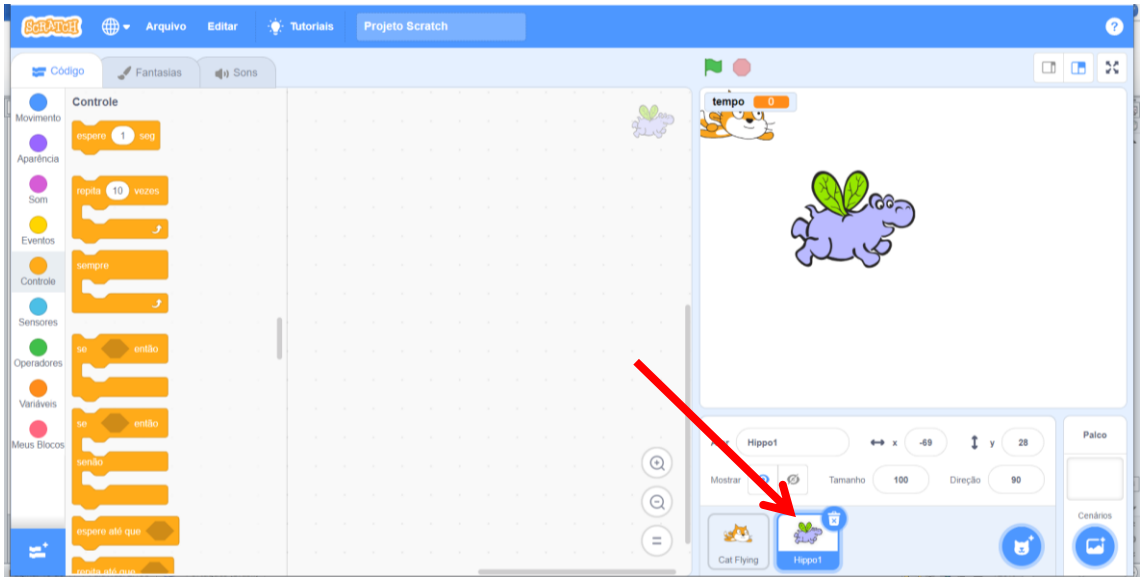


Faça um teste e verifique se o contador está funcionando.

Agora vamos escolher outro ator para ser o “inimigo”.

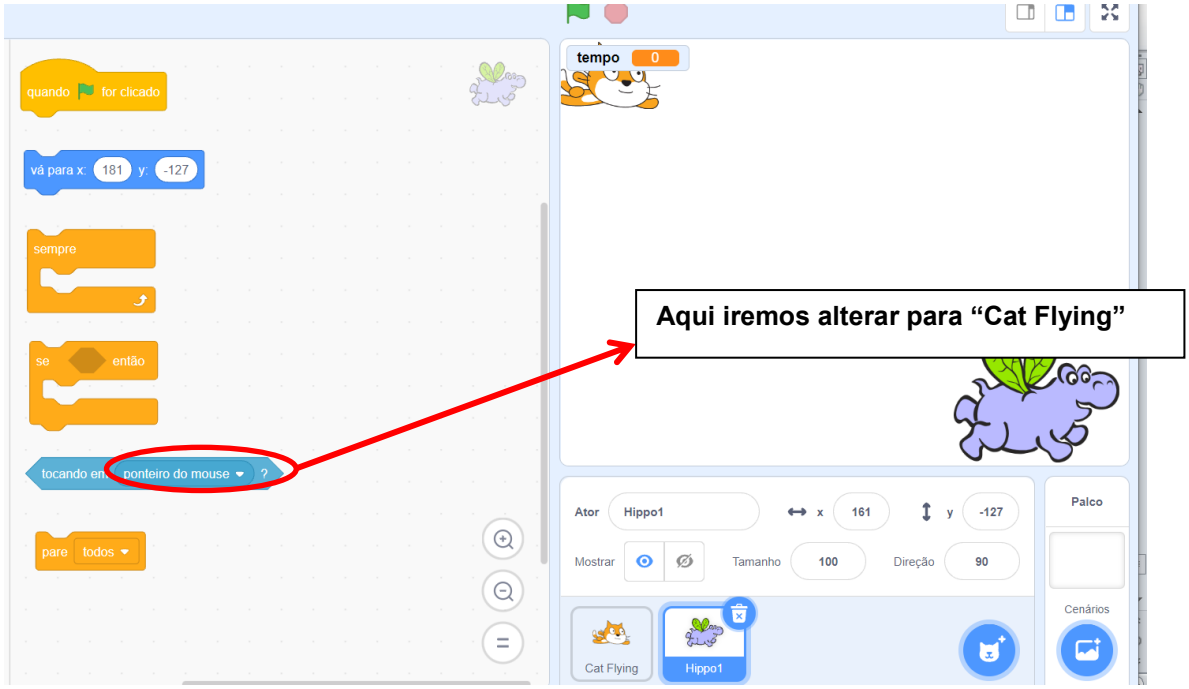


Escolha o ator “Hippo 1”:

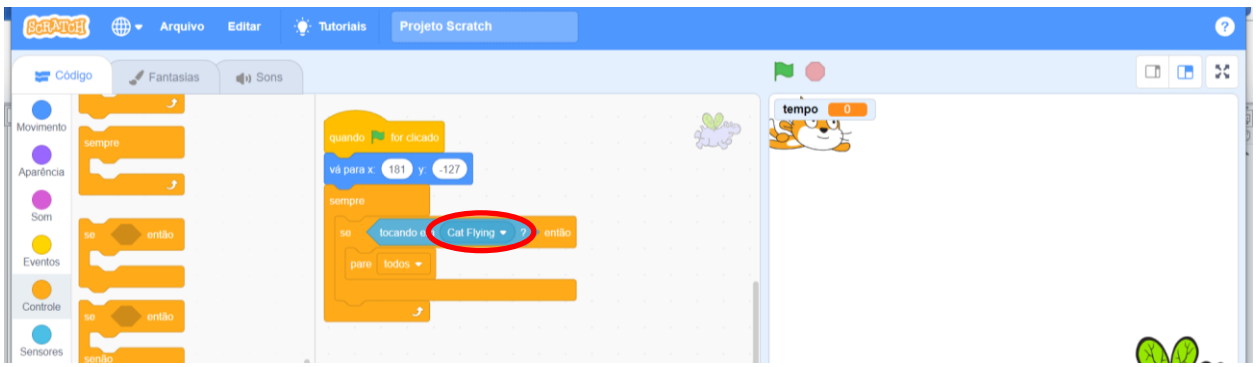


Observe que está selecionado o ator Hippo 1, e é nele que vamos inserir os **DOIS** novos blocos de programação.

O primeiro bloco:



Sua tela deve ficar assim:



No segundo bloco vamos usar os comandos:

Sua tela deve ficar assim:

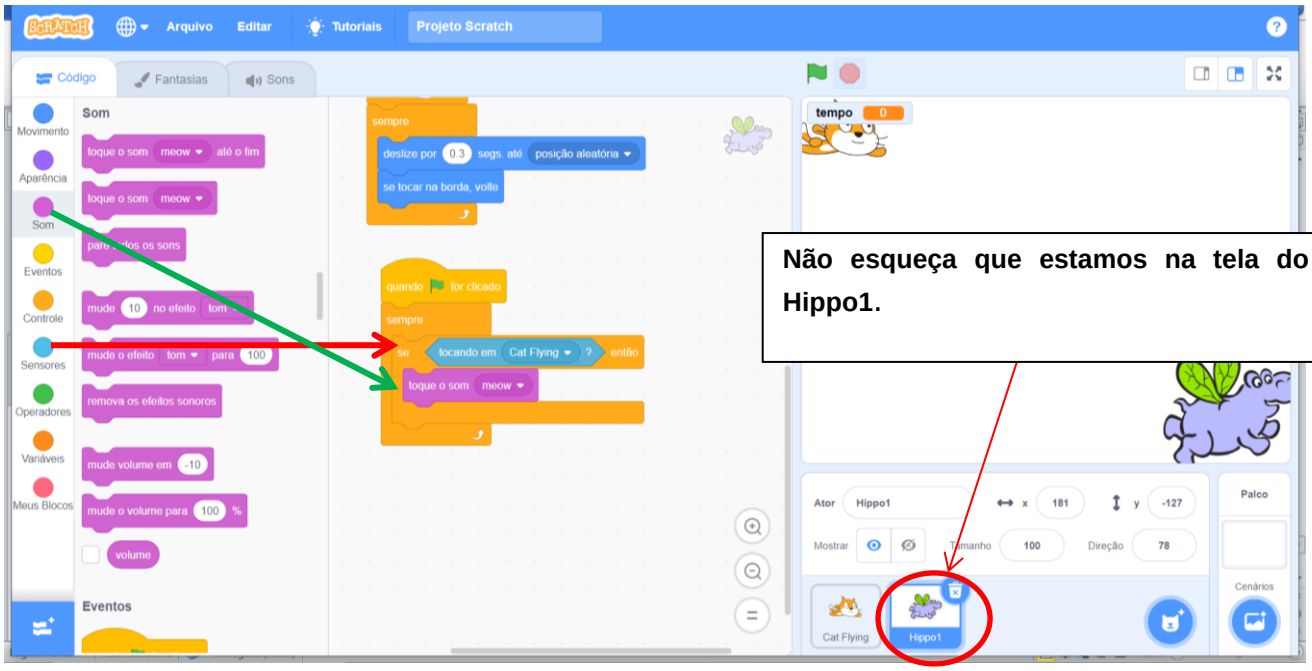
O jogo já está pronto e você pode realizar o teste, entretanto é possível incrementar e deixá-lo mais bonito e robusto.

Vamos acrescentar um som toda vez que nosso personagem cat flying tocar no “inimigo”.

Na mesma tela do Hippo1, vamos acrescentar um bloco para que o som seja emitido.

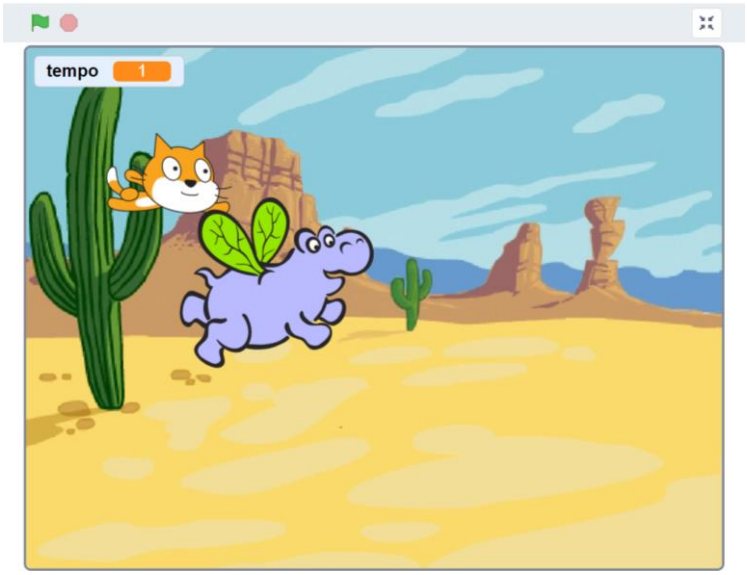
Não se esqueça de alterar “tocando em” para nosso personagem Cat Flying

Sua tela deve fica assim:



Não esqueça que estamos na tela do Hippo1.

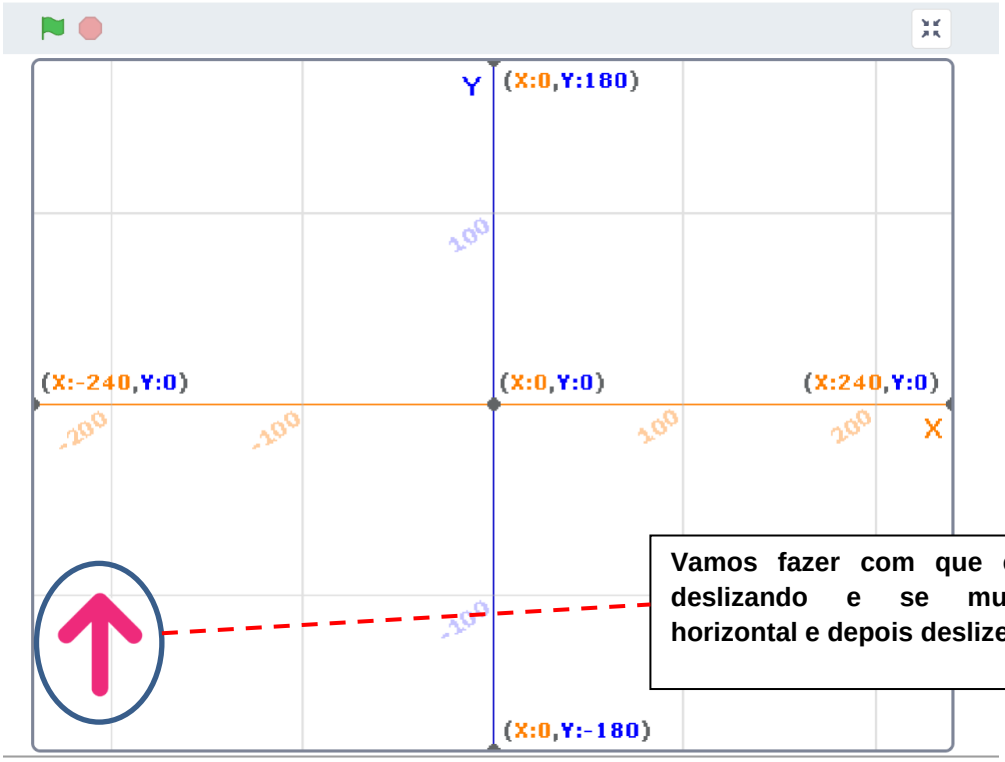
Agora pode acrescentar um cenário de sua escolha e pronto, nosso jogo está finalizado. Teste e divirta-se!

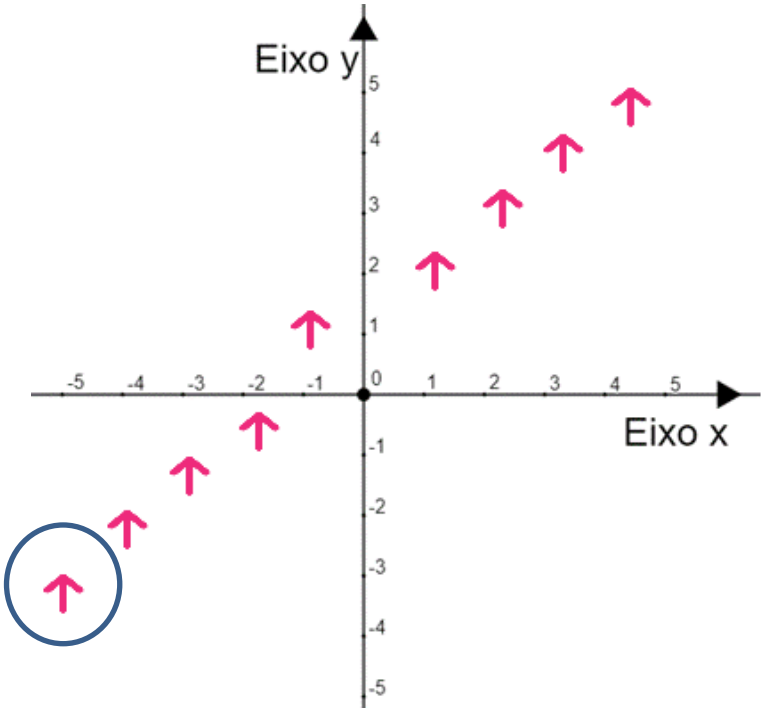


Aula 6: O objetivo desta aula é aprendermos como se cria clones infinitos de um mesmo ator.

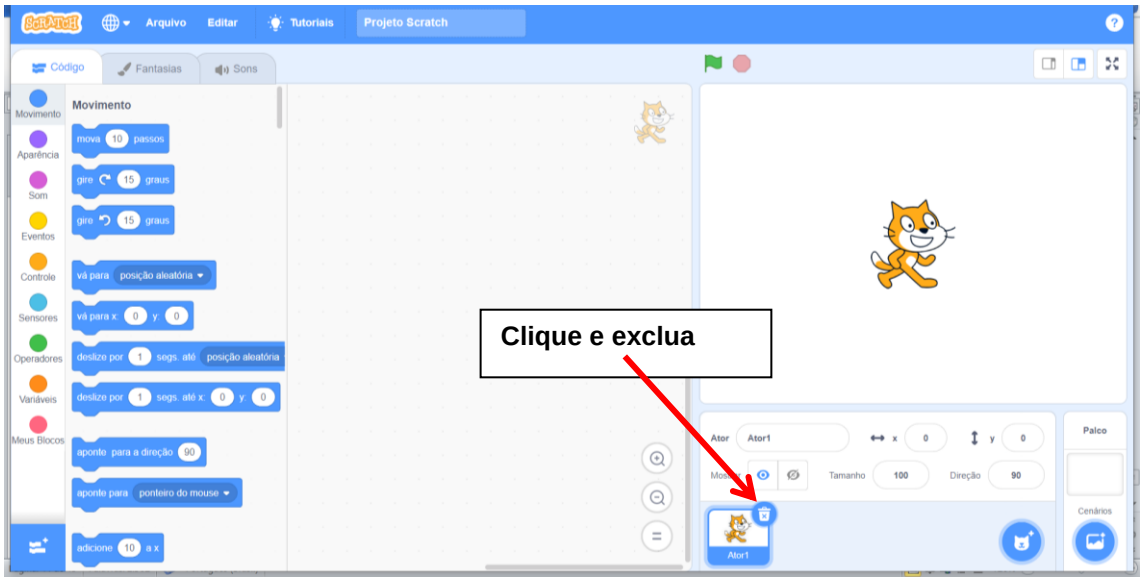
Primeiro faremos com que o ator crie clones na horizontal (eixo X). E a na aula 7, faremos os clones na vertical (eixo Y). É muito importante entender esses conceitos para continuidade do curso.

A ideia nessa aula é fazer com que os clones apareçam no eixo x (horizontal) e deslizem no eixo y (vertical). Ficou confuso? Calma, vamos analisar a ideia num desenho.

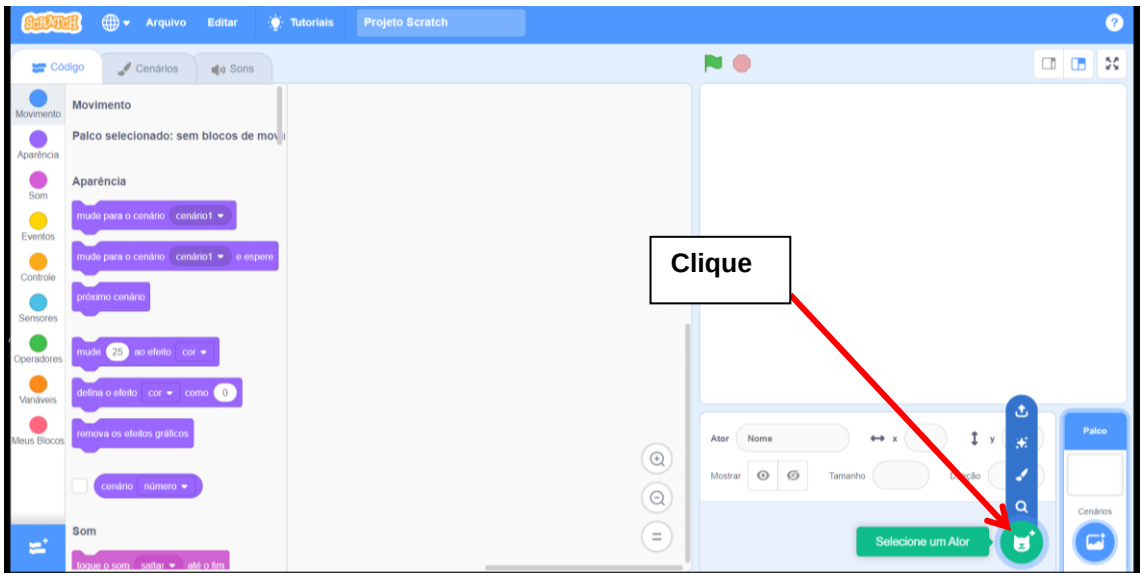




Abra o Scratch e exclua o ator 1.

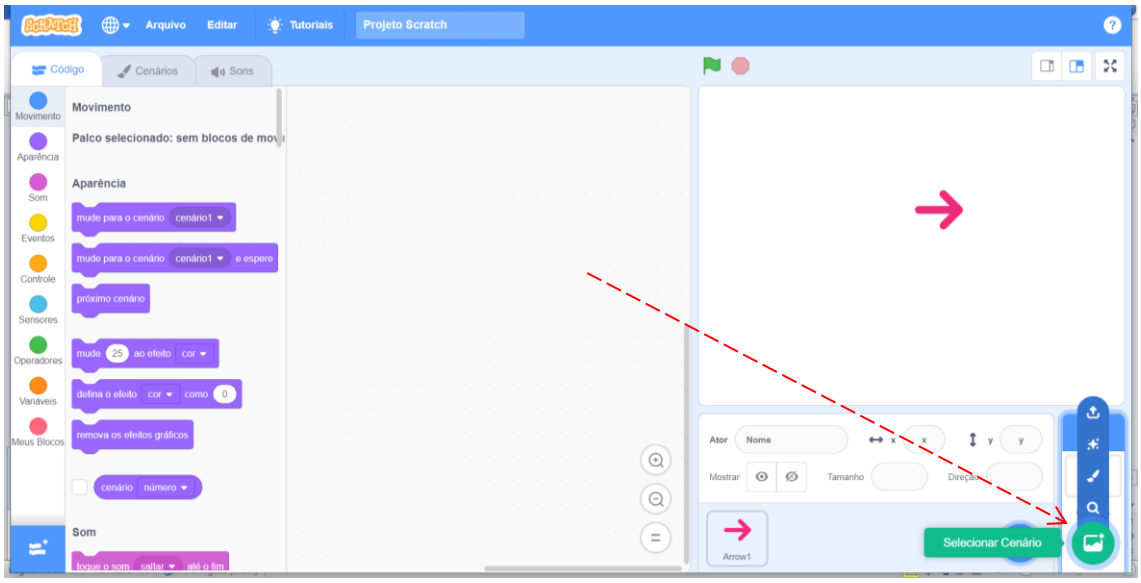


Acrescente um novo ator:

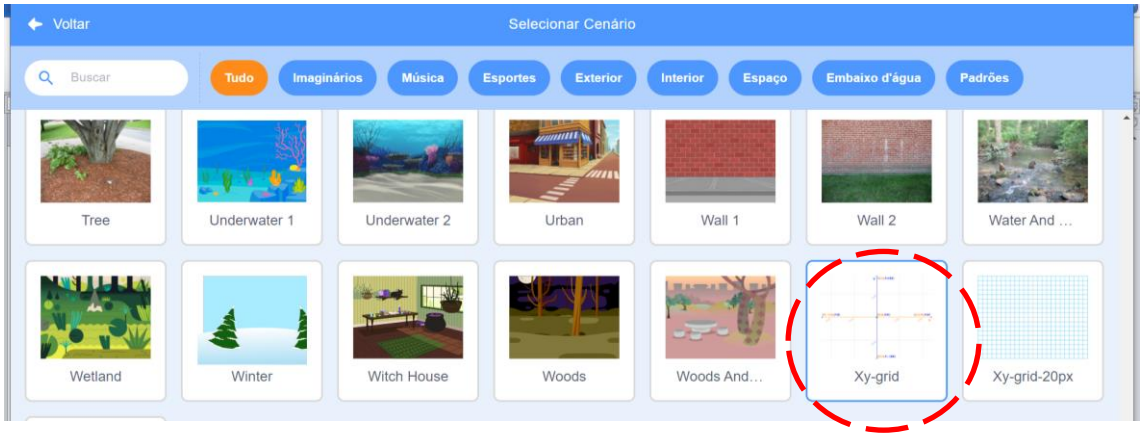


Escolha o ator ARROW 1

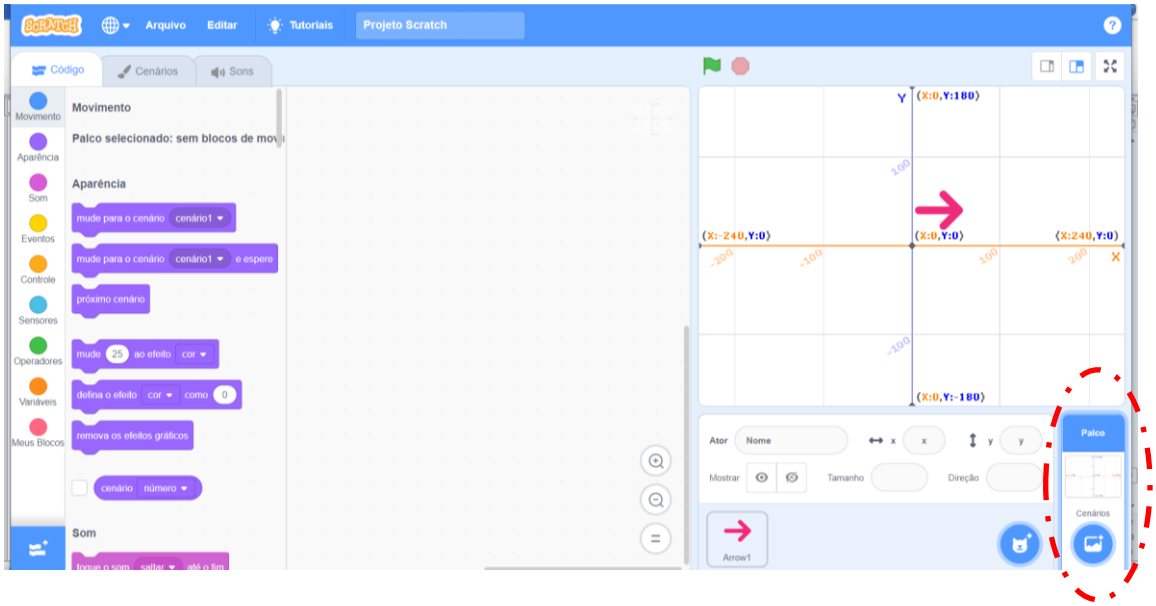
Agora vamos escolher um cenário.



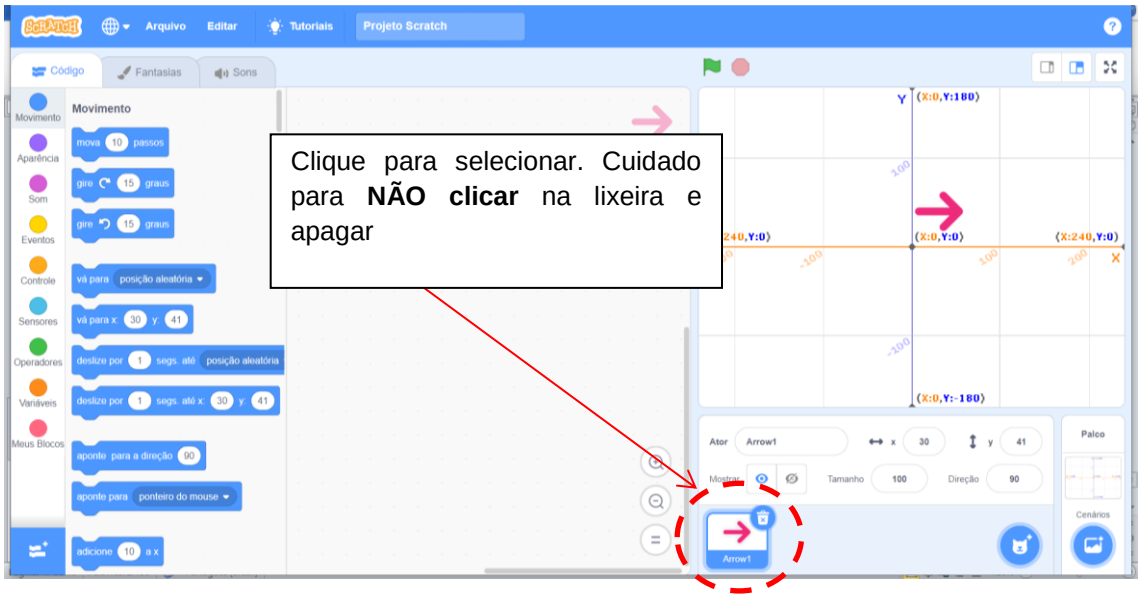
Selecione o cenário Xy-grid



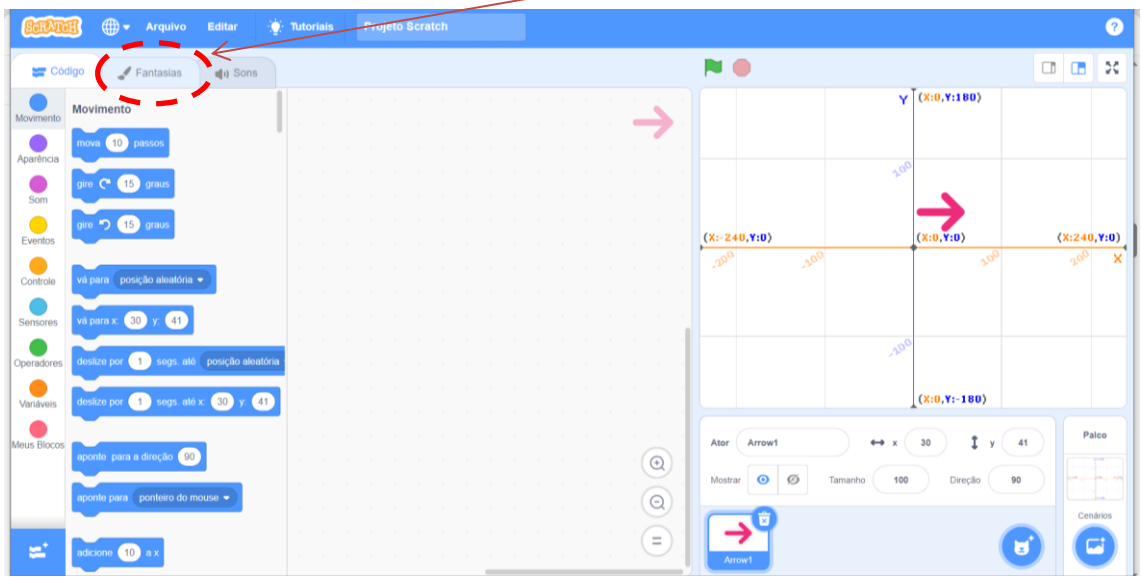
Sua tela deve ficar assim:



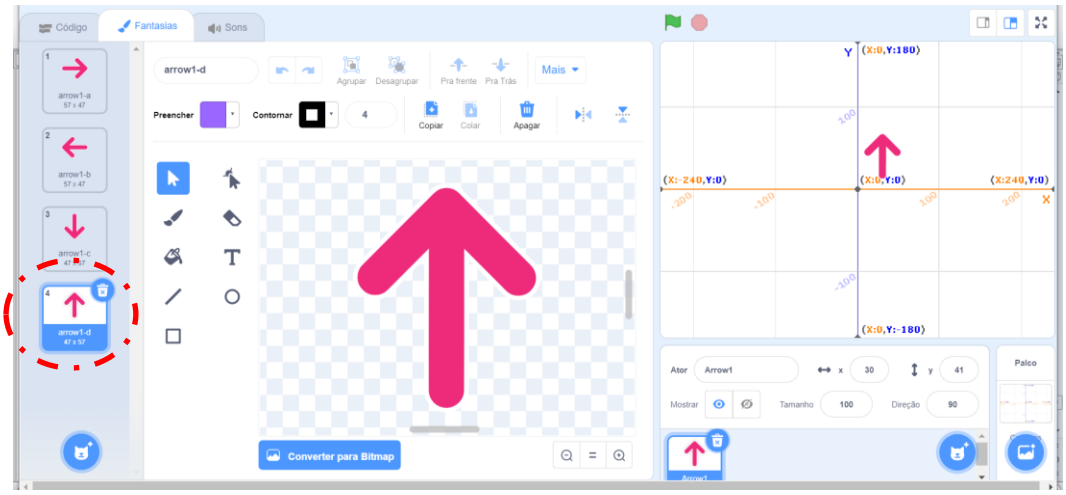
Muita atenção agora. Vamos programar o ator 1 (Arrow1), portanto, devemos selecioná-lo.



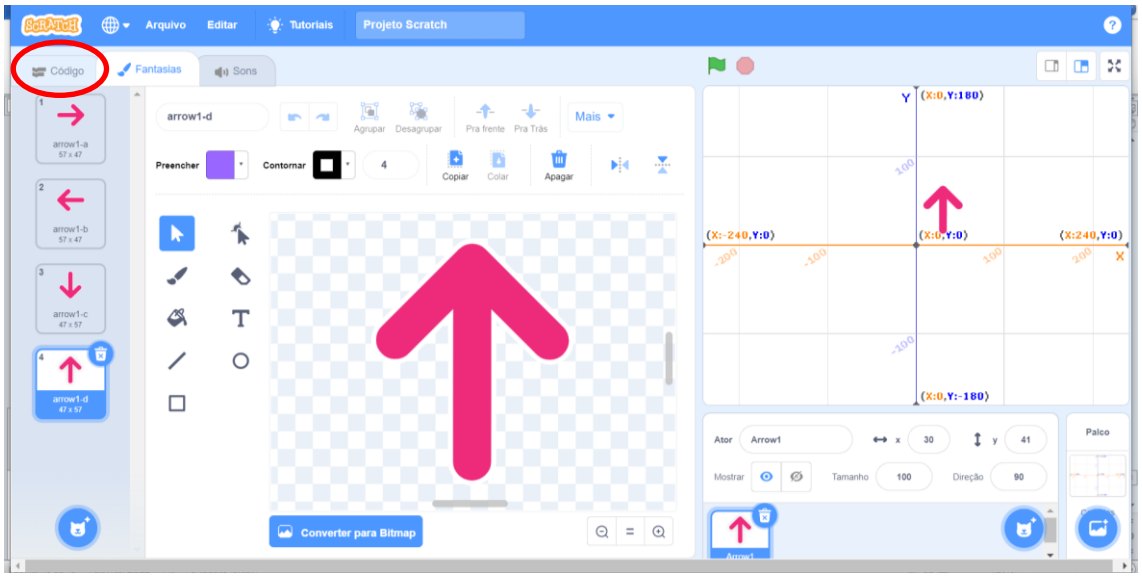
Agora que você selecionou o ator, clique na aba Fantasia:



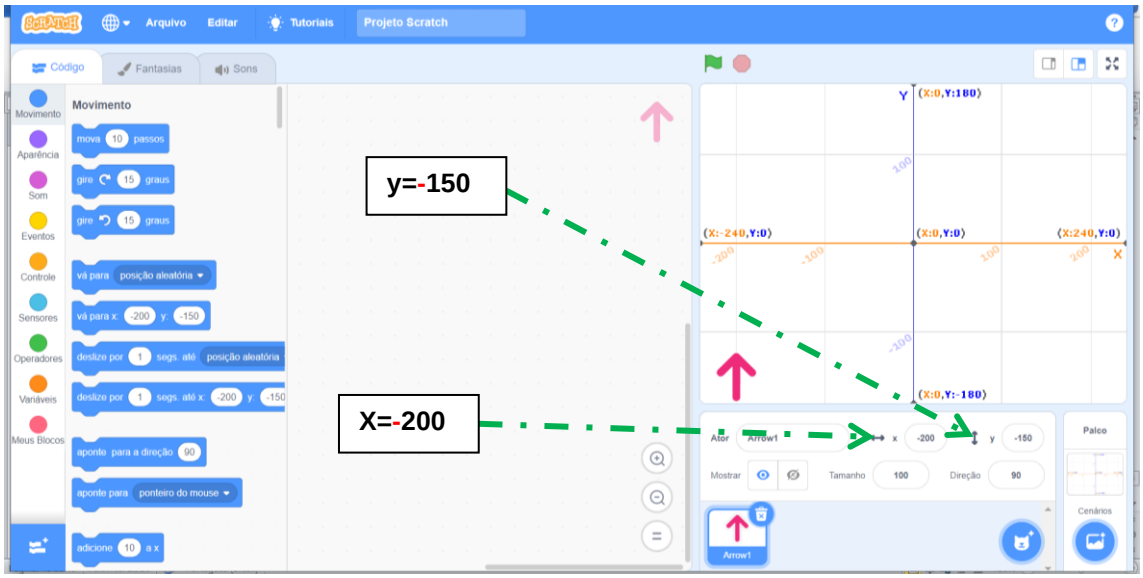
Selecione a fantasia em que a seta está voltada para cima:



Volte para o código:

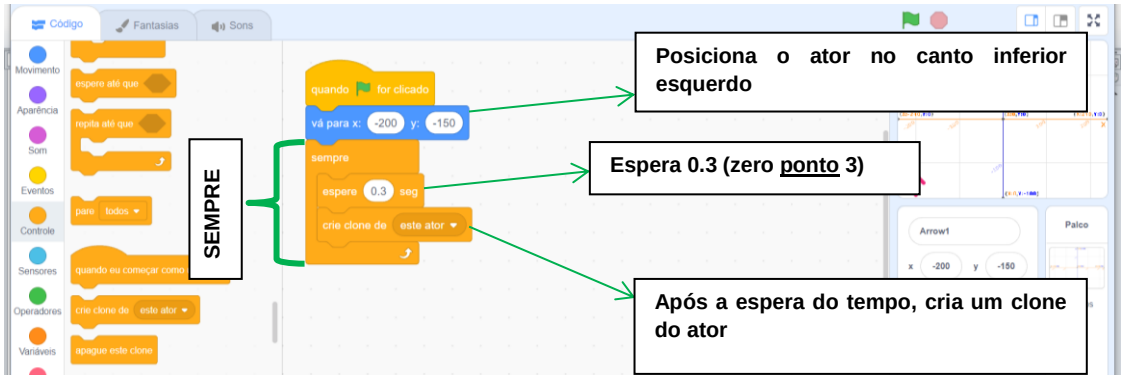


Vamos posicionar o ator no canto inferior esquerdo da tela com x = -200 e y = -150:

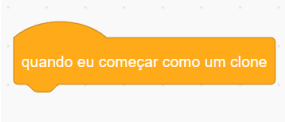


Agora vamos começar a programação. A ideia **não é apenas copiar o código, e sim entender a lógica da programação**.

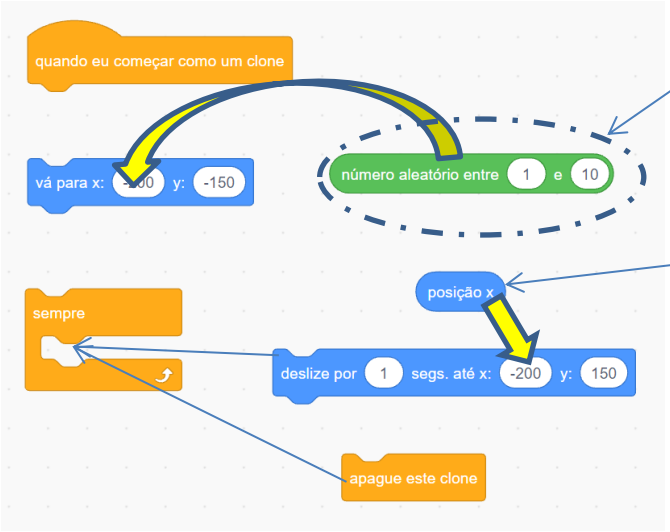
Primeira parte: criando os clones



Segunda parte: Após criar os clones, o que fazer com eles?

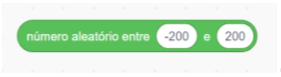


Vamos usar estes comandos:



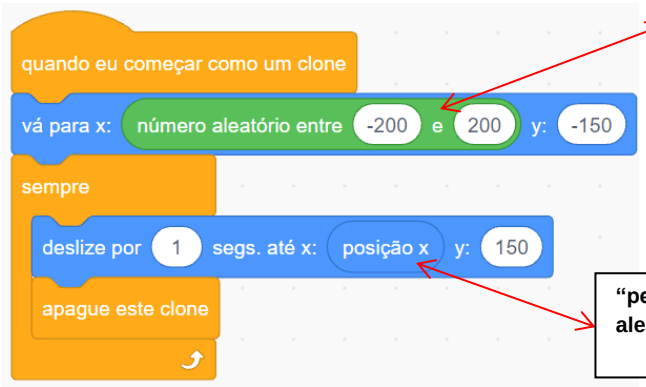
Esse comando escolhe um número aleatório entre dois escolhidos. Vamos usá-lo para que na coordenada x, sejam escolhidos números entre -200 e 200. Isso fará com que os clones sempre apareçam na horizontal, isto é, entre x = -200 e x =200.

Esse comando "posição x" faz com que o programa "pegue" o valor x que foi escolhido aleatoriamente no <u>comando anterior:</u>



.

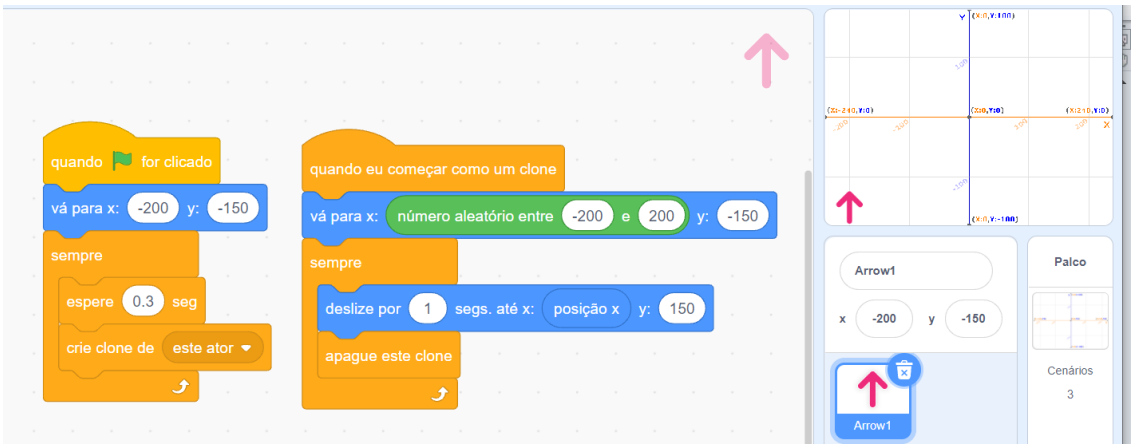
Não copie apenas. Procure entender a lógica da programação!



Escolhe um valor para o x entre -200 e 200. Isso faz com que o clone apareça no eixo horizontal x entre esses valores.

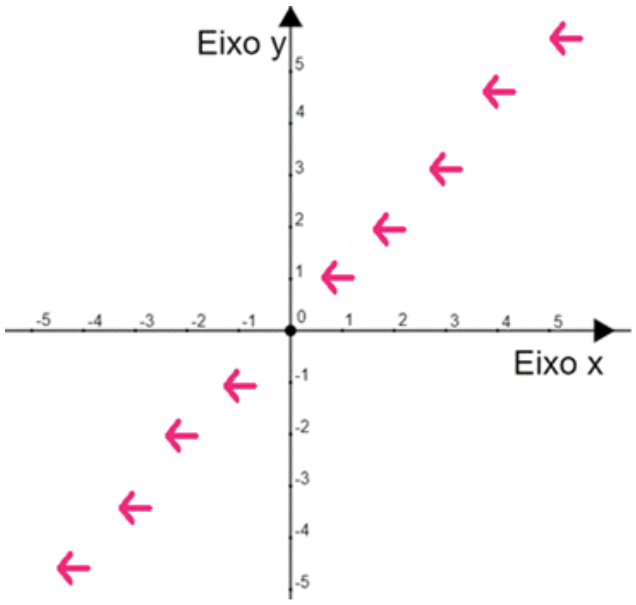
"pega" o valor de x que foi escolhido aleatoriamente no comando anterior

Sua tela deve ficar assim:



Agora inicie a animação e observe que o ator está criando clones no eixo x, e esses clones, estão deslizando na vertical (direção eixo y).

Aula 7: Caro (a) estudante, nesta aula você vai fazer uma animação semelhante a aula 6, mas agora os clones devem deslizar na direção horizontal (eixo y).



Na próxima página tem os blocos que você vai utilizar para realizar a animação. É muito semelhante ao que fizemos na aula 6, apenas vamos alterar alguns valores para que os clones deslizem na horizontal.

Ator: Arrow 1

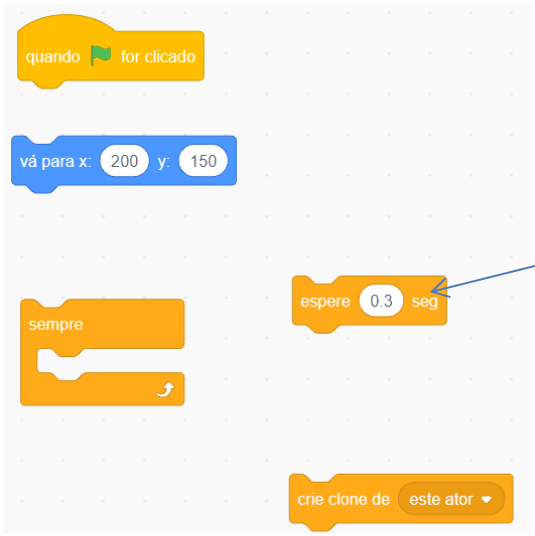
Cenário: xy-grid

Não se esqueça de ir em "fantasias" e deixar o ator apontando para o lado esquerdo da tela.

X= 200

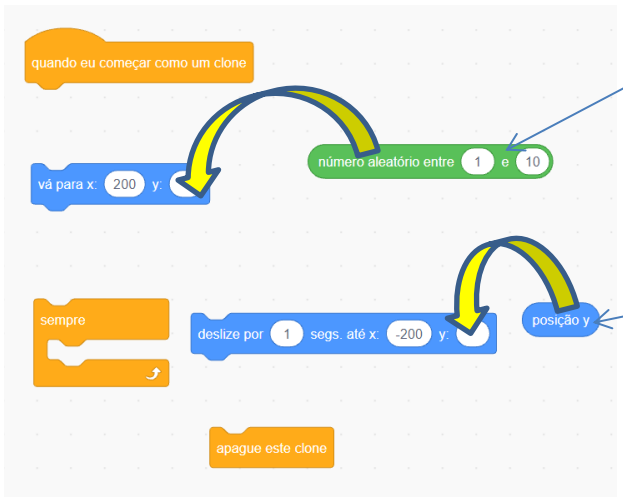
y= 150

PRIMEIRA PARTE:



Não se esqueça de colocar ponto

SEGUNDA PARTE:

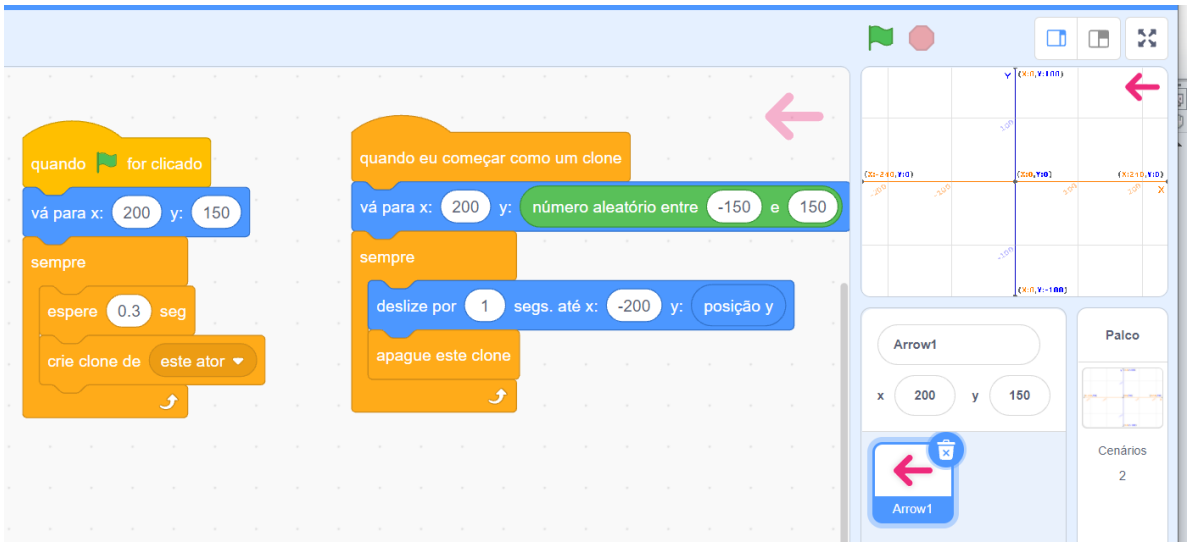


Esse comando escolhe um número aleatório entre dois escolhidos. Vamos usá-lo para que na coordenada y, sejam escolhidos números entre -150 e 150. Isso fará com que os clones sempre apareçam na vertical, entre y = -150 e y = 150.

"pega" o valor de y que foi escolhido aleatoriamente no comando anterior

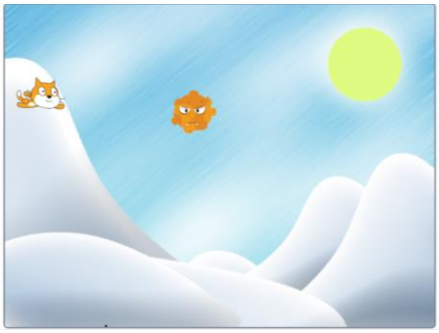
Tente fazer sozinho. Tem a resposta na próxima página, mas é importante que você tente realizar atividade.

RESPOSTA:

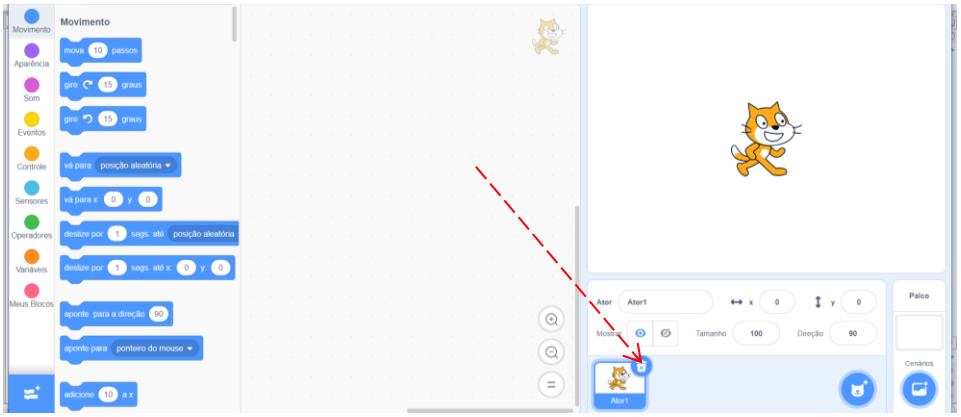


Faça o teste e verifique se a animação está correta.

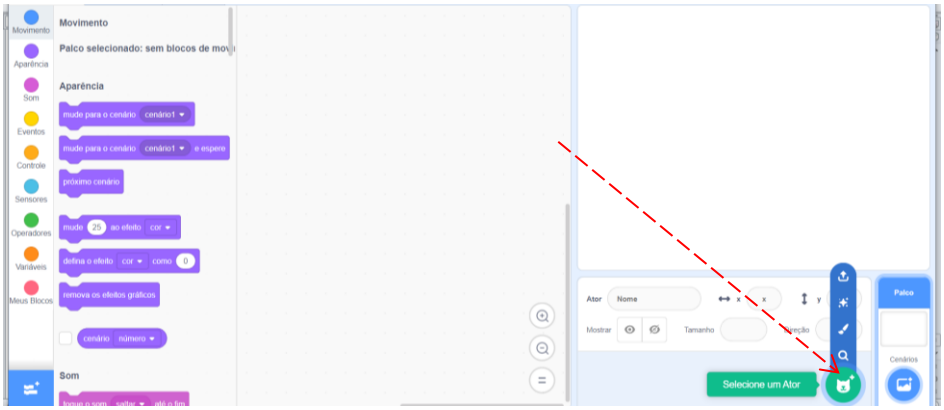
Aula 8: Nesta aula vamos simular o jogo Flappy Bird, para isso, vamos utilizar o que foi discutido na aula 7, isto é, criação de clones e o deslizando desses na direção horizontal.

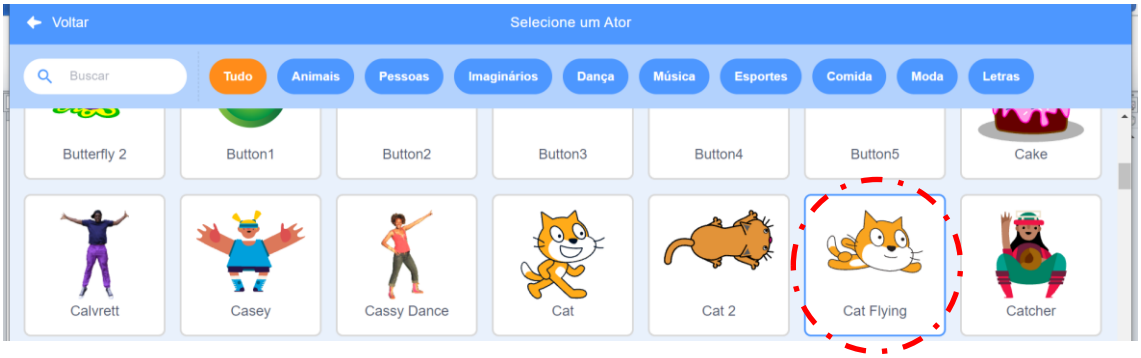


Abra o Scratch e exclua o ator 1:

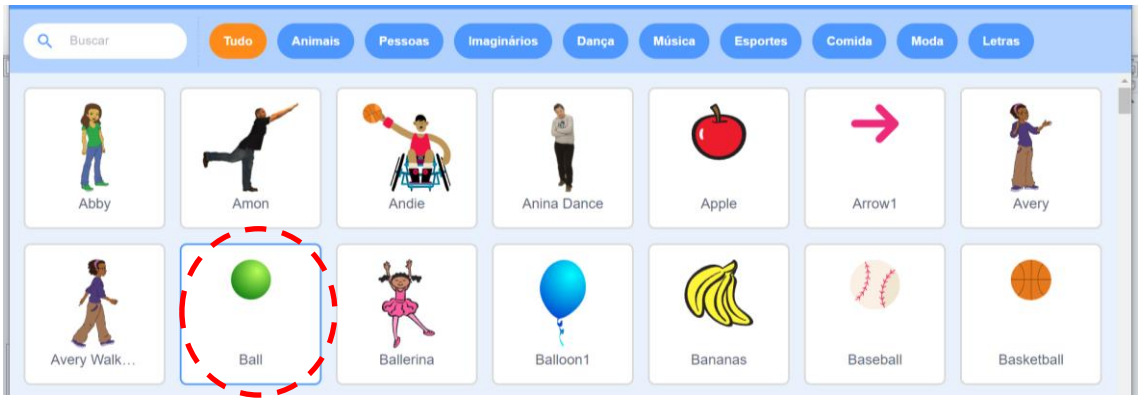


Vamos selecionar o ator Cat Flying

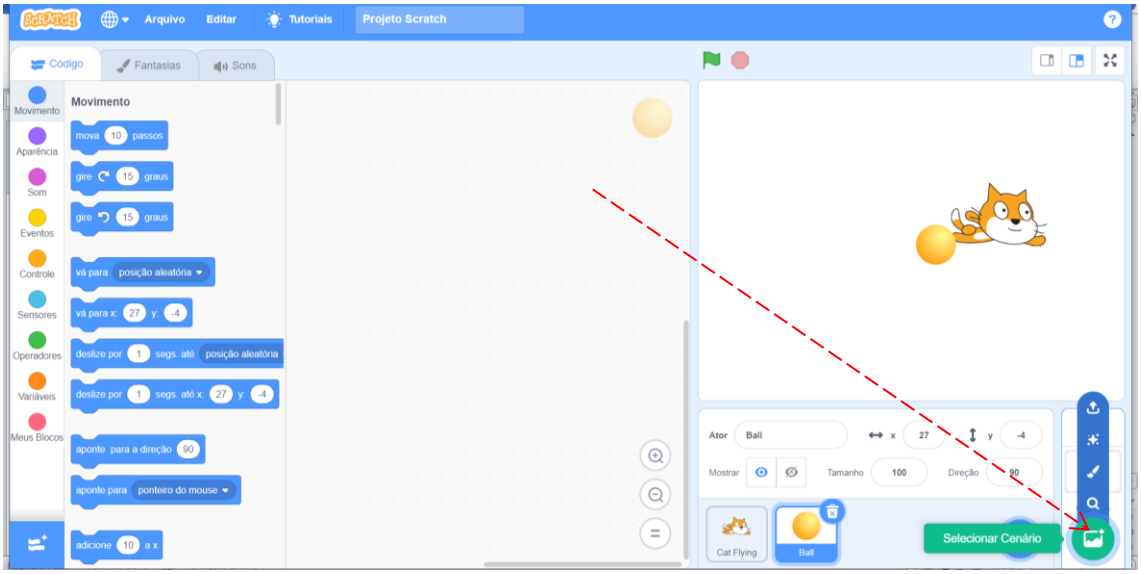




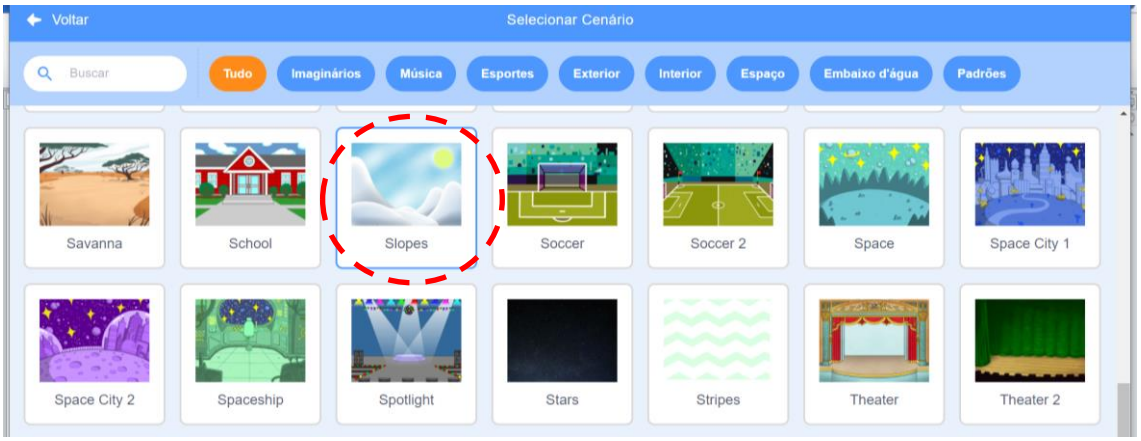
Repita o procedimento e selecione o segundo ator (BALL):



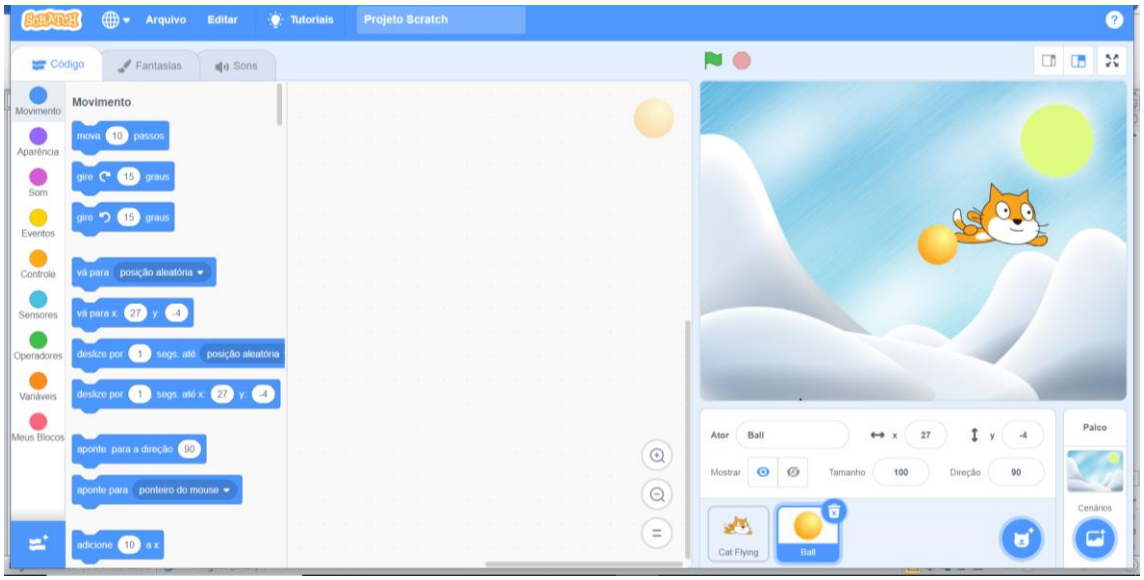
Agora vamos selecionar um cenário:



Selecione o cenário Slopes:



Sua tela deve ficar assim:



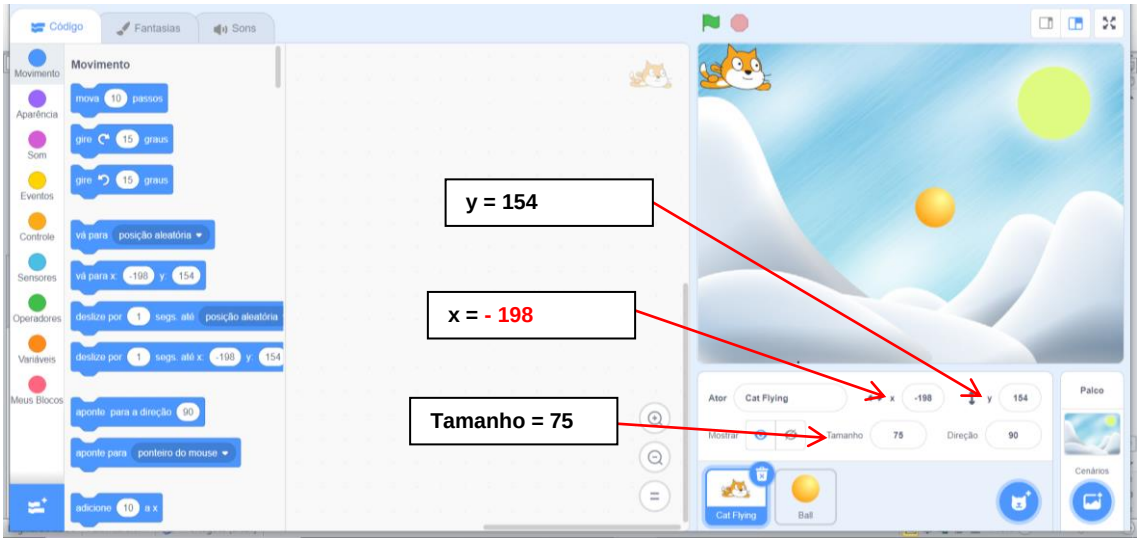
Agora temos 3 objetos: 2 atores e um cenário. Não se esqueça de selecionar que desejamos programar.

Para trabalharmos com o ator 1 (cat flying) devemos selecioná-lo.

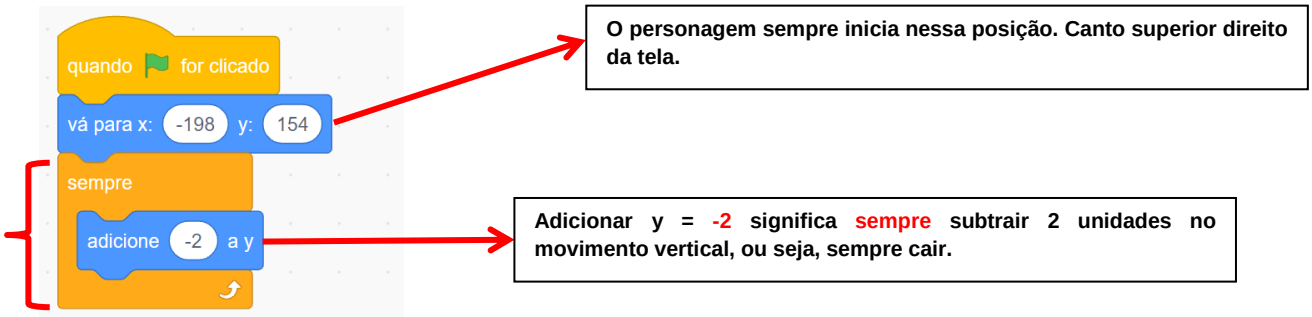
Para trabalharmos com o ator 2 (Ball) devemos selecioná-lo.

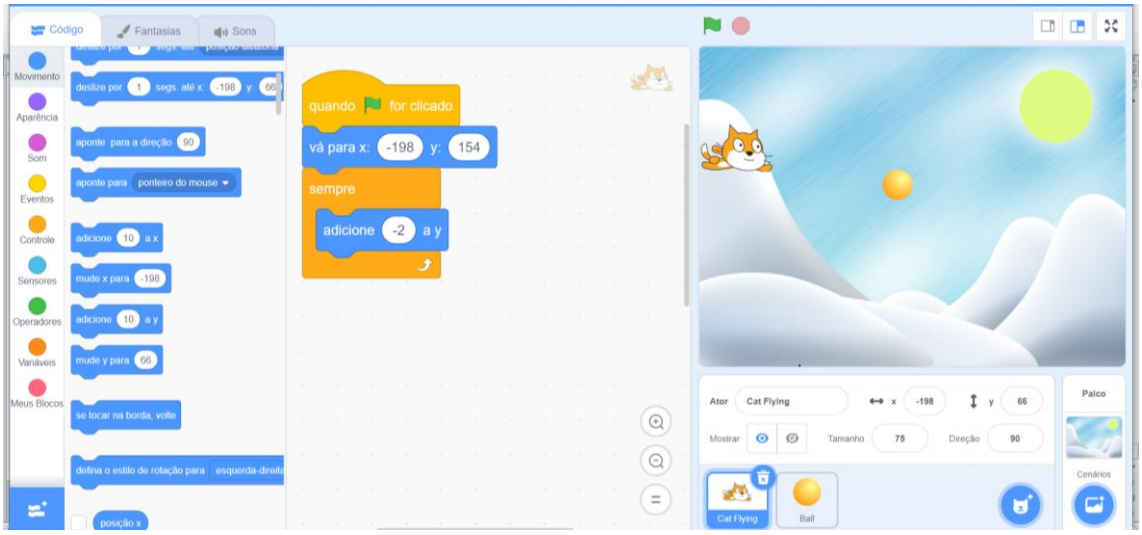
Para trabalharmos com o cenário devemos selecioná-lo.

Primeiro vamos programar o ator 1 (Cat flying). Vamos selecioná-lo e começar a programação. A ideia é que ele fique sempre caindo na tela e quando acionarmos a tecla “espaço” faça um leve movimento de subida.

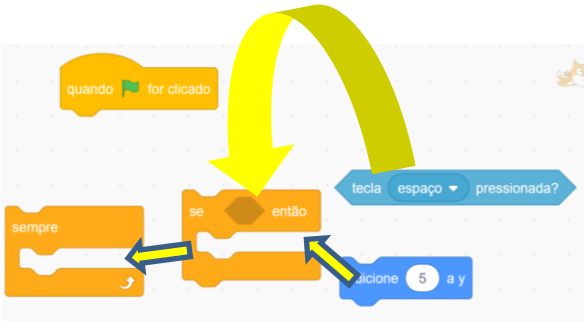


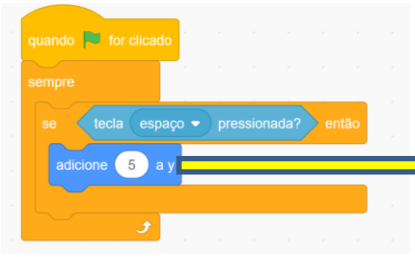
No primeiro bloco de programação precisamos que ator sempre fique caindo. Cair significa que vamos deslocar o ator na direção vertical (eixo y).





Agora precisamos que ao pressionar a tecla “espaço” o personagem faça um leve movimento de subida.





Por que devemos adicionar 5 positivo a y?

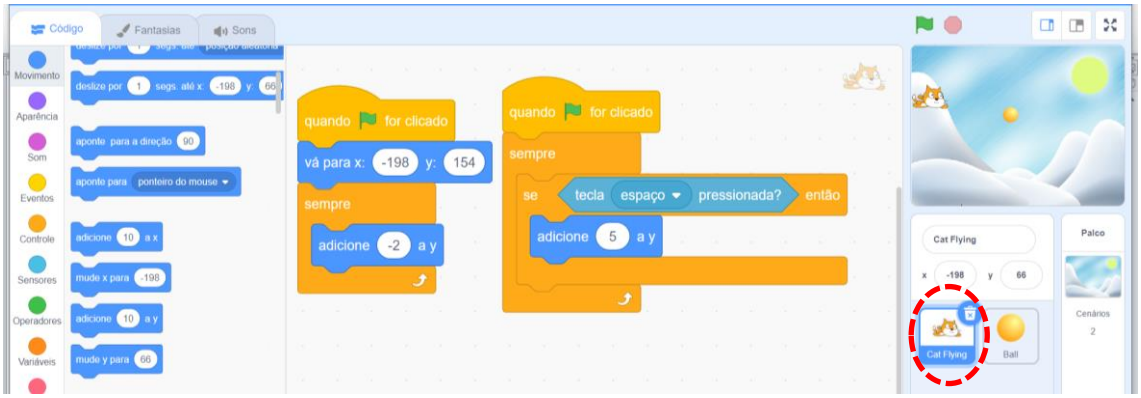
Lembre-se que o ator está sempre caindo porque estamos subtraindo 2 unidades. Para que ele tenha um leve movimento de subida, devemos adicionar um valor maior de 2, matematicamente estamos fazendo o seguinte cálculo:

- 2 sempre

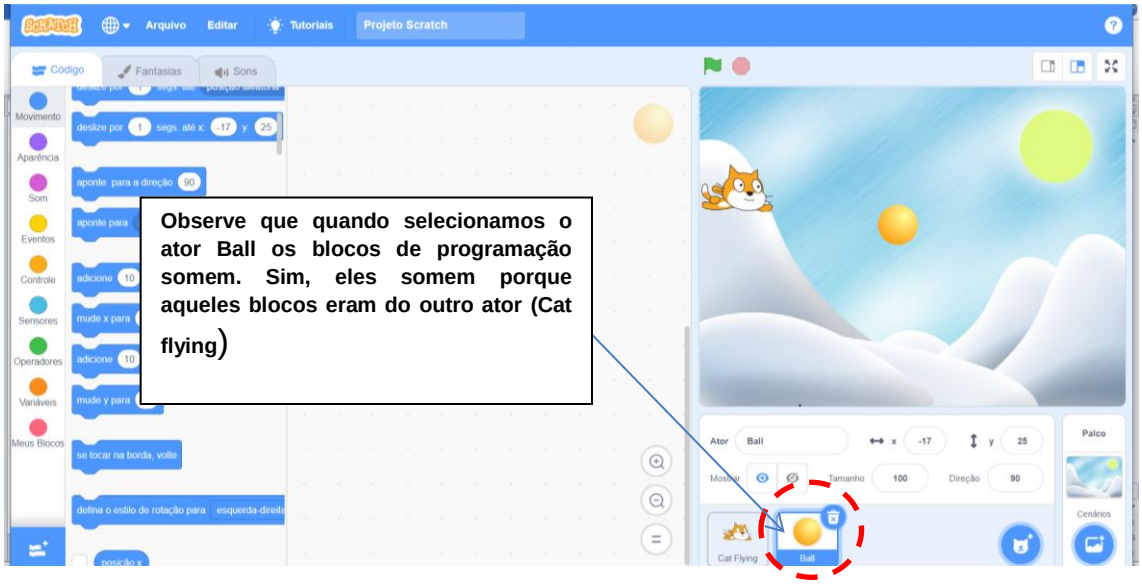
Quando acionamos a tecla “espaço” adicionamos 5:

5 - 2 = +3 (sobe 3 unidades)

Sua tela deve ficar assim:

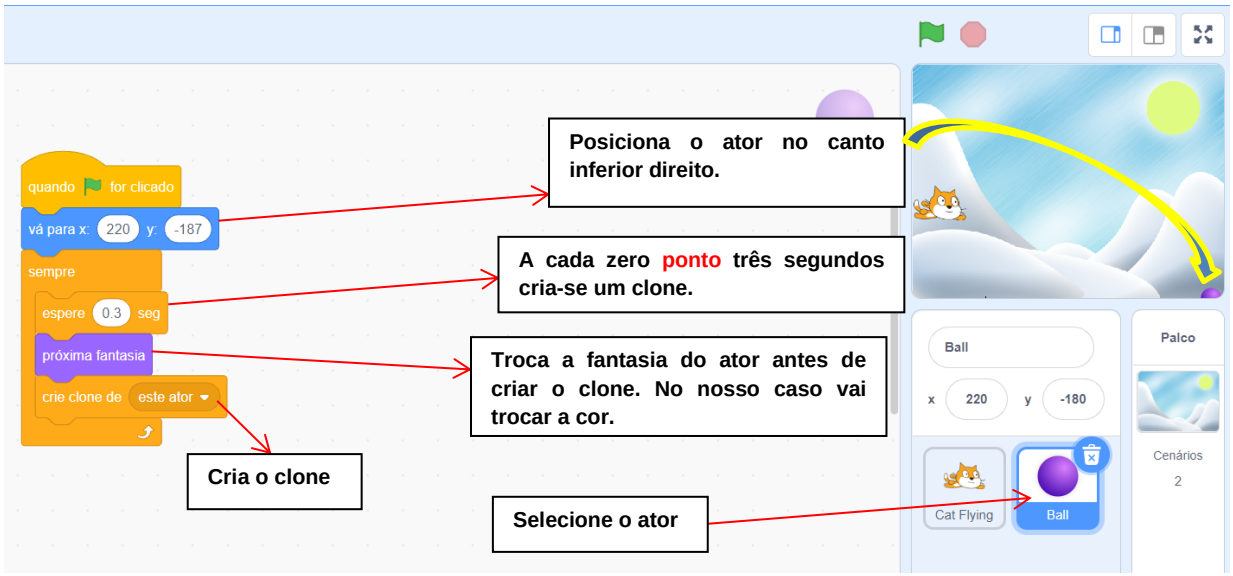


Agora vamos **selecionar** o segundo ator (ball) para iniciarmos sua programação:

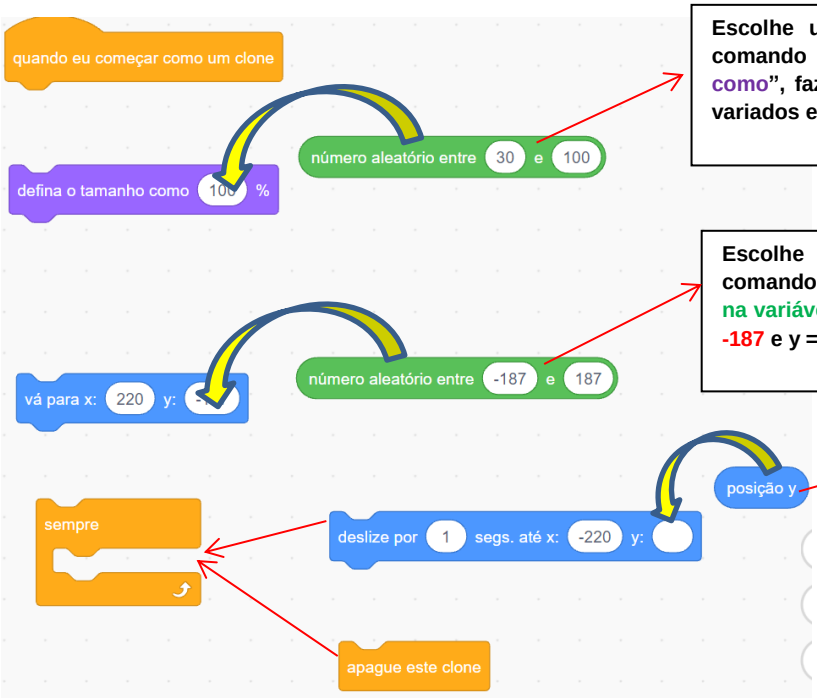


Vamos precisar criar clones que deslizem na horizontal, e se tocar no cat flying o jogo tem que parar. Serão 3 blocos de comando para esse ator.

Primeiro vamos criar os clones:



E depois dos clones criados, o que queremos? É isso que o segundo bloco irá fazer. **Entenda a lógica da programação!**

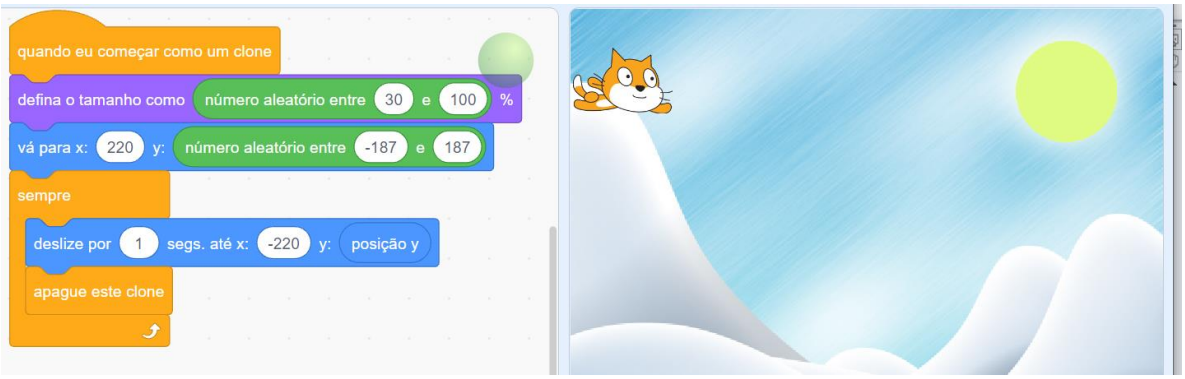


Escolhe um número aleatório entre 30 e 100. Esse comando encaixado no comando "defina tamanho como", faz com que os clones apareçam em tamanhos variados entre 30% e 100%.

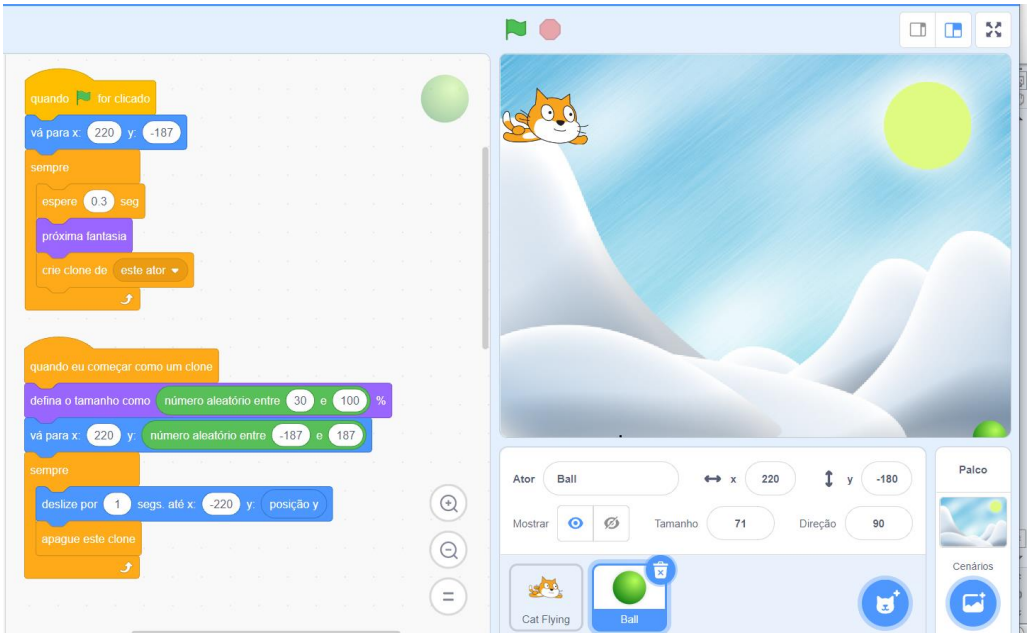
Escolhe um número aleatório entre -187 e 187. Esse comando encaixado no comando "vá para x: 200 e y:__", na variável y, faz com que os clones apareçam entre y = -187 e y = 187, ou seja, na vertical.

O comando "posição y" pega o último valor de y atribuído no comando anterior, isto é, o y entre -187 e 187.

Segundo Bloco:

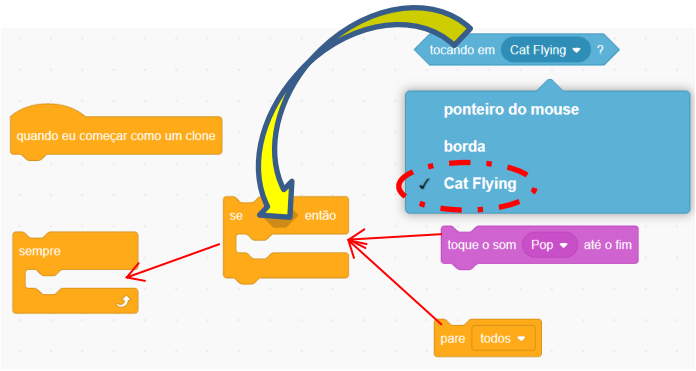


Sua tela deve ficar assim:

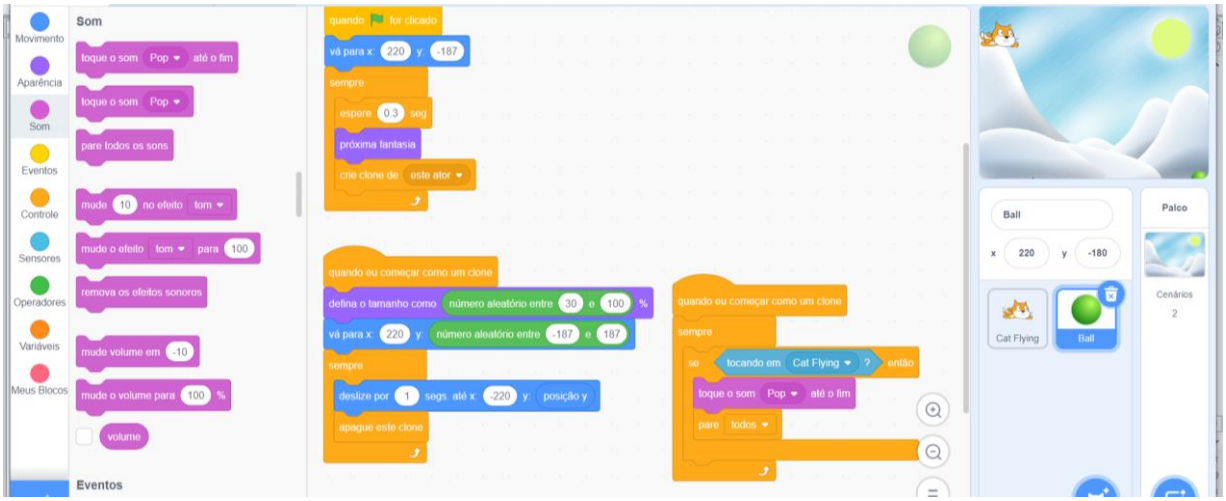


Ainda falta um bloco que vai parar o jogo quando o ator Cat Flying tocar em um dos clones.

Comandos:



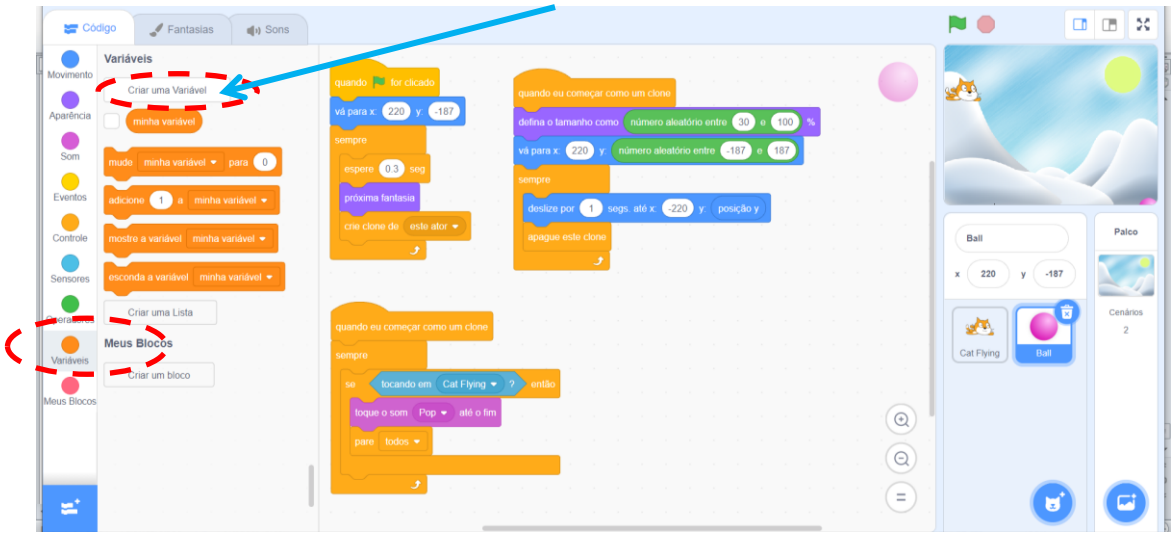
Sua tela deve ficar assim:



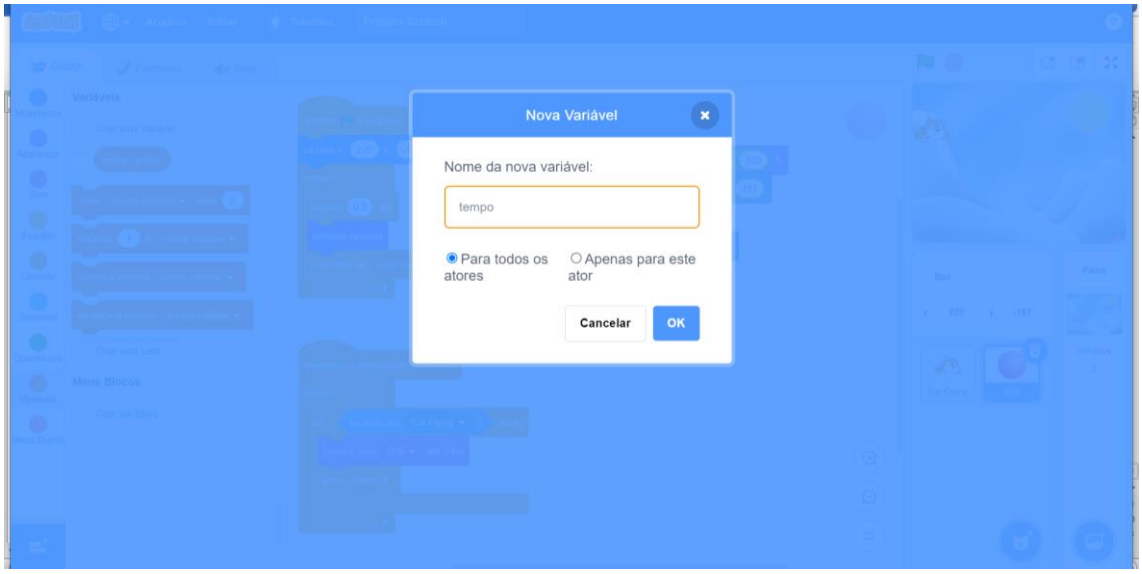
FAÇA O TESTE DO JOGO!

É possível aprimorarmos o jogo e colocarmos um contador de tempo. Para isso vamos acrescentar um quarto bloco.

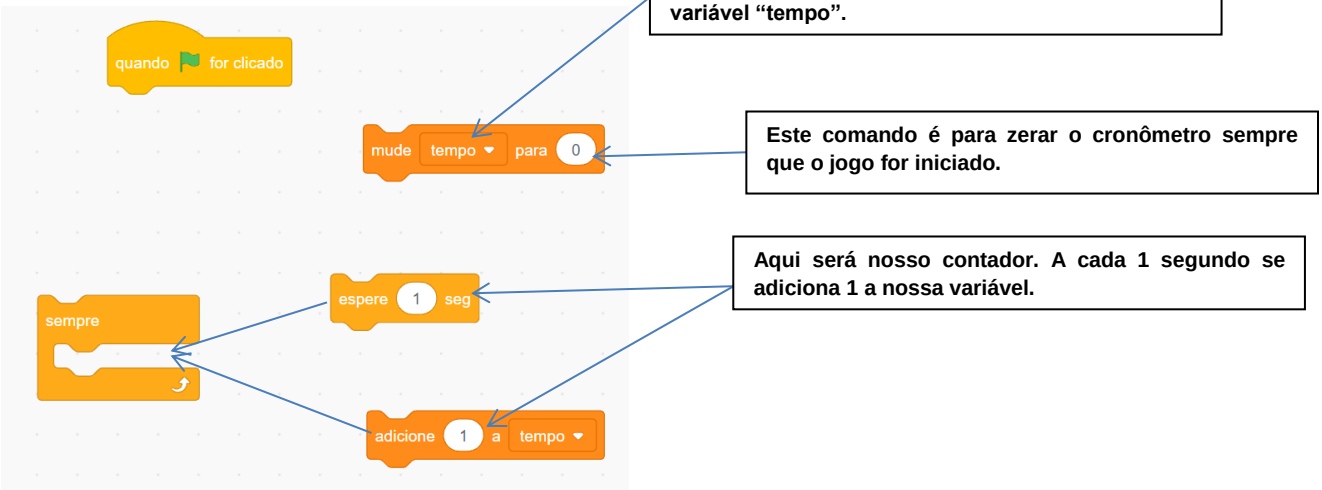
Vamos permanecer nesse mesmo ator (Ball). Primeiro vamos criar uma variável para o tempo. Clique em “Criar uma Variável”



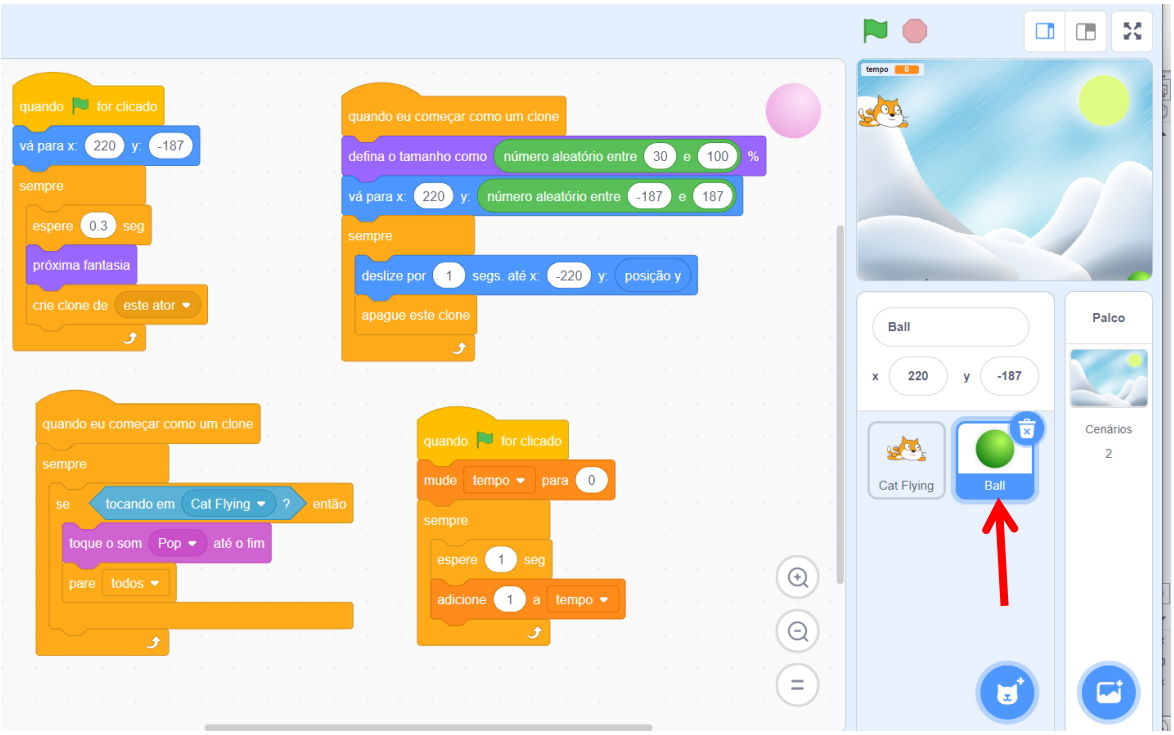
Nomeie a variável como “tempo” e clique em OK



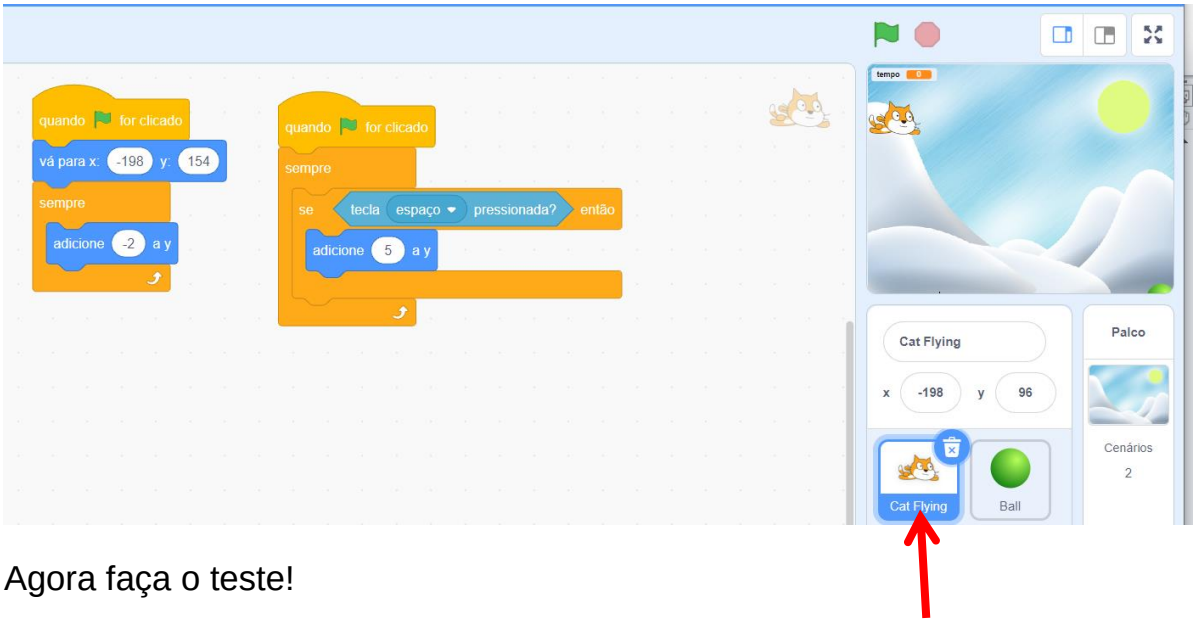
Agora vamos montar nosso quarto bloco:



Sua tela com os quatro blocos no ator Ball deve ficar assim:

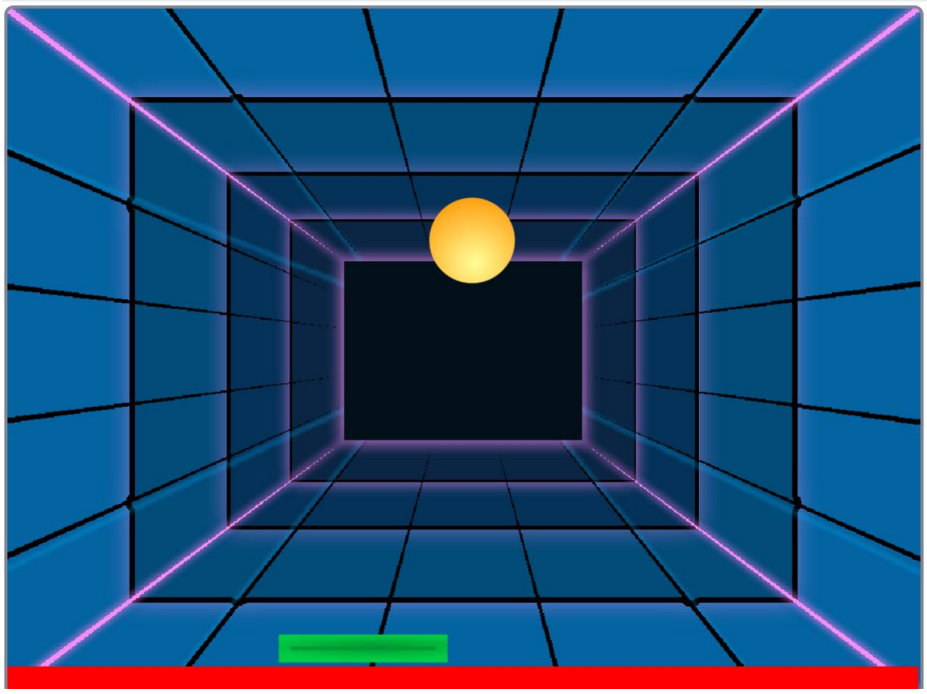


E com os dois blocos no ator Cat Flying sua tela deve fica assim:

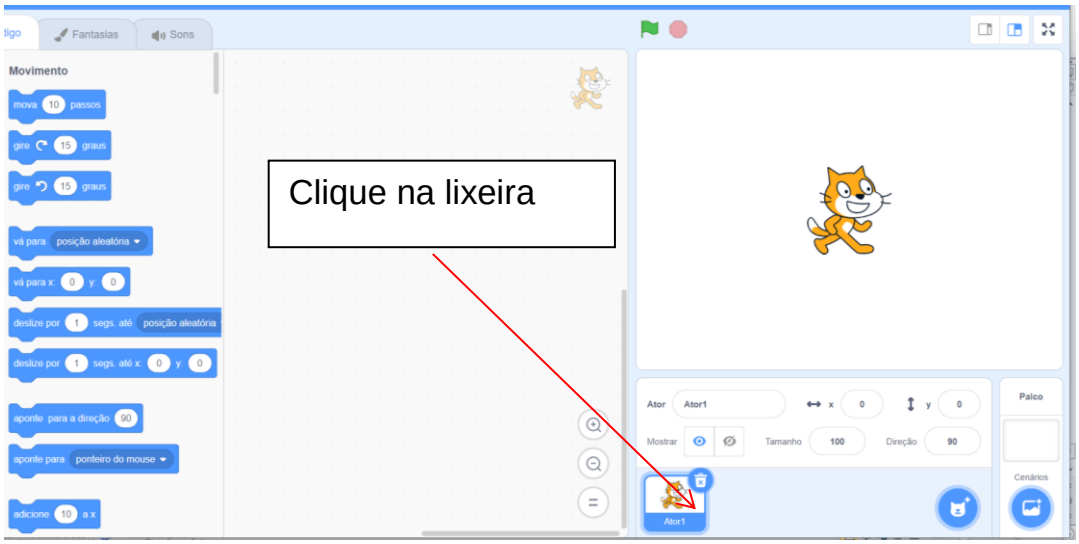


Agora faça o teste!

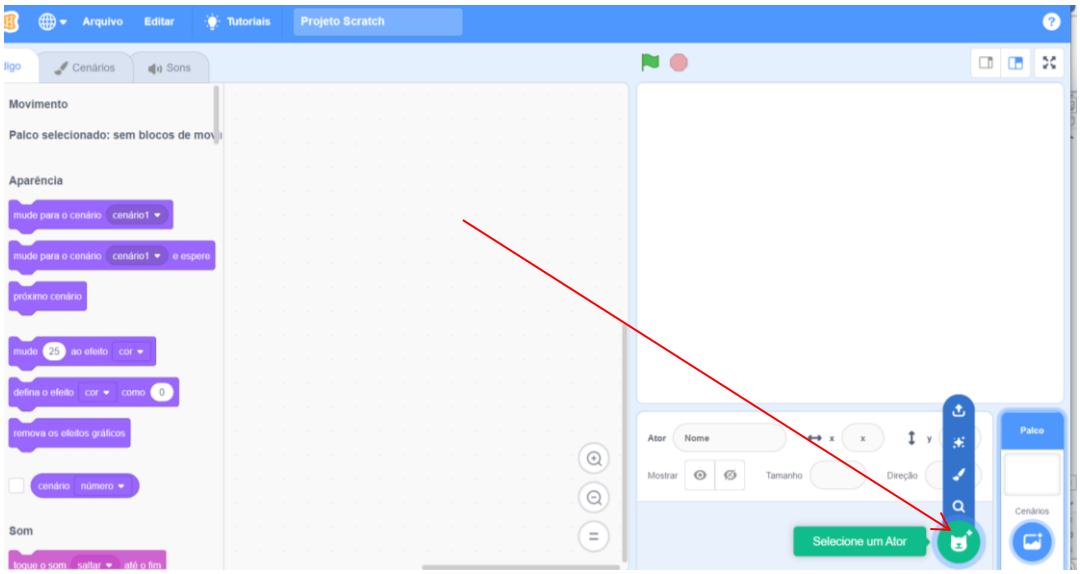
Aula 9: Nesta aula vamos criar o jogo Pong.



Abra o Scratch e exclua o ator:

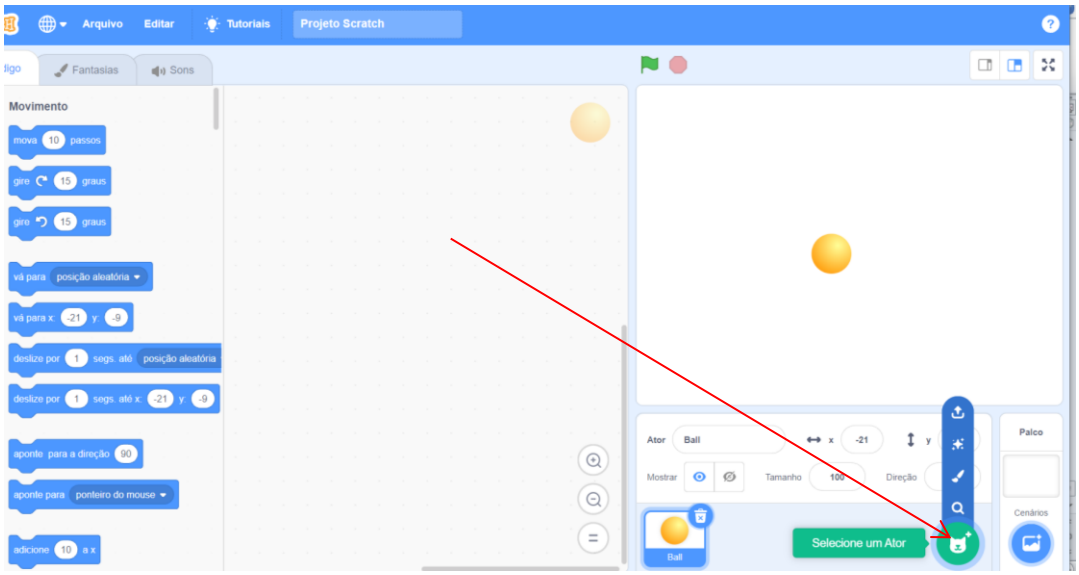


Agora vamos selecionar três atores:

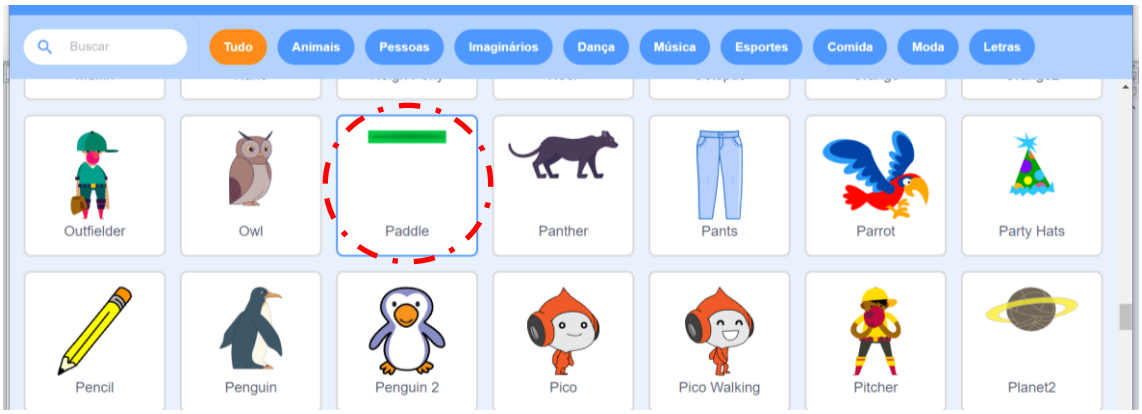


Selecione Ball:

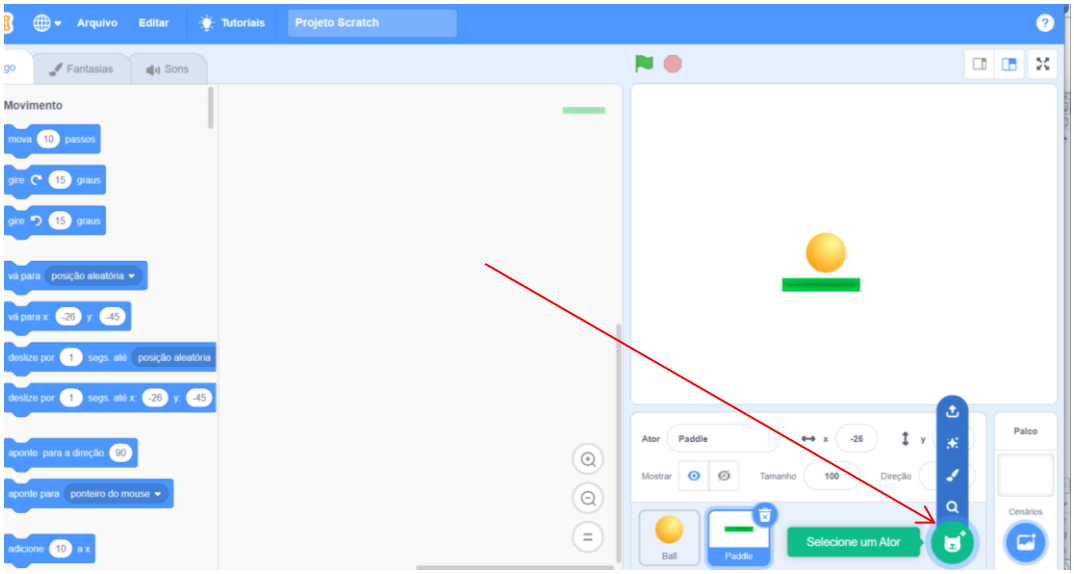
Repita o procedimento e vamos selecionar o segundo ator:



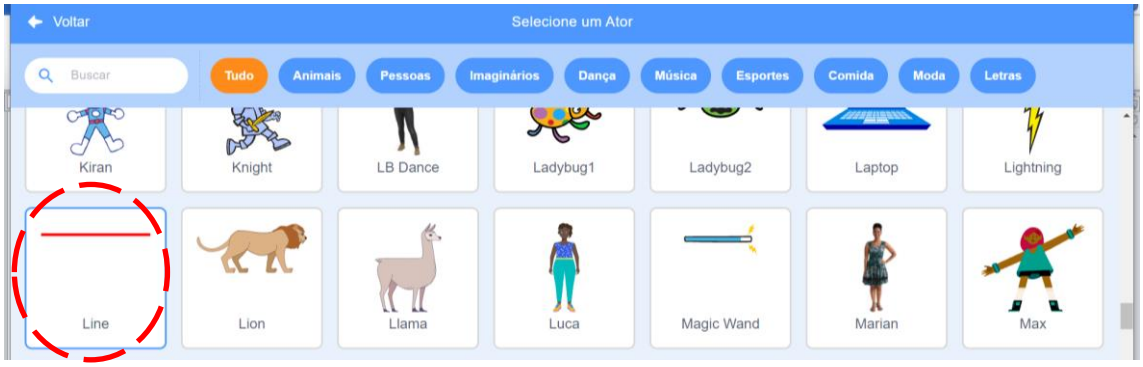
Selecione Paddle



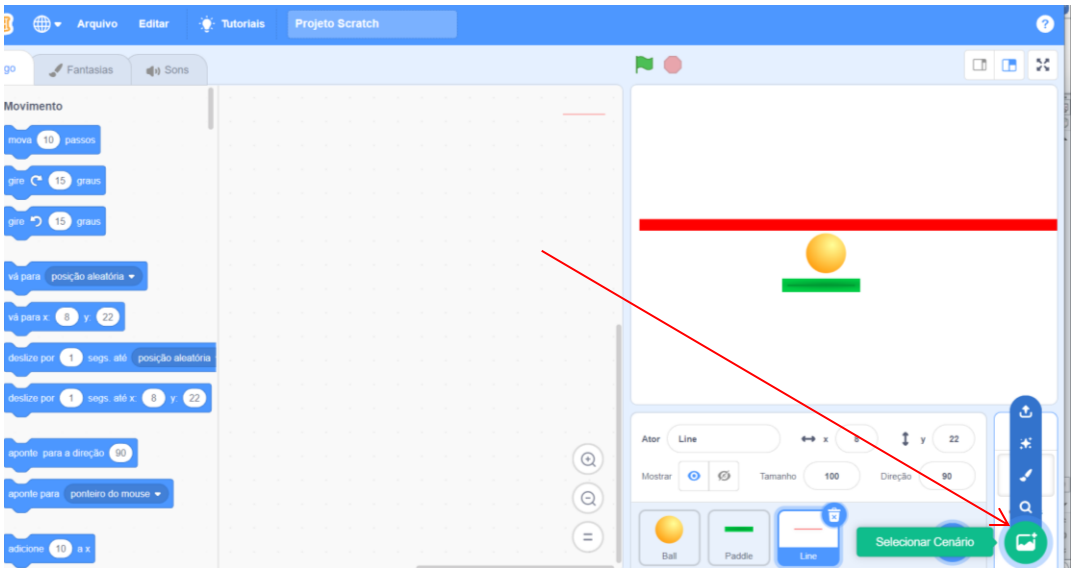
Repita o procedimento para selecionarmos o terceiro ator:



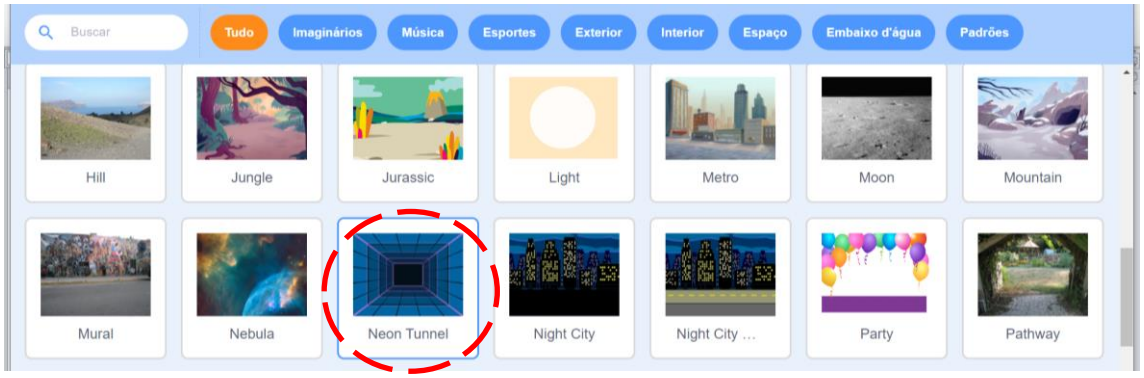
Selecione LINE



Agora vamos selecionar um cenário:

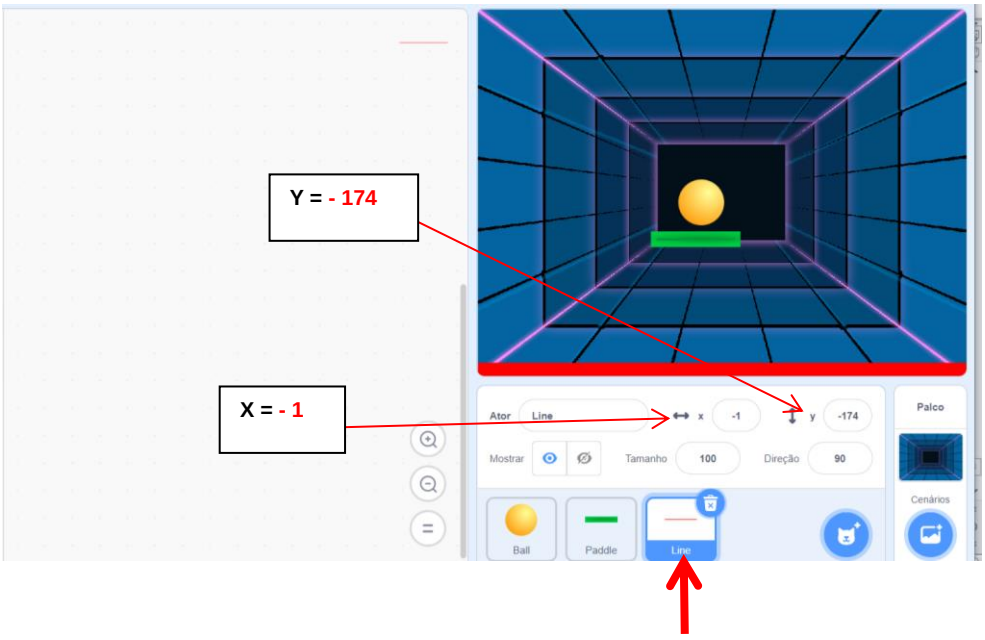


Selecione o cenário NEON TUNEL

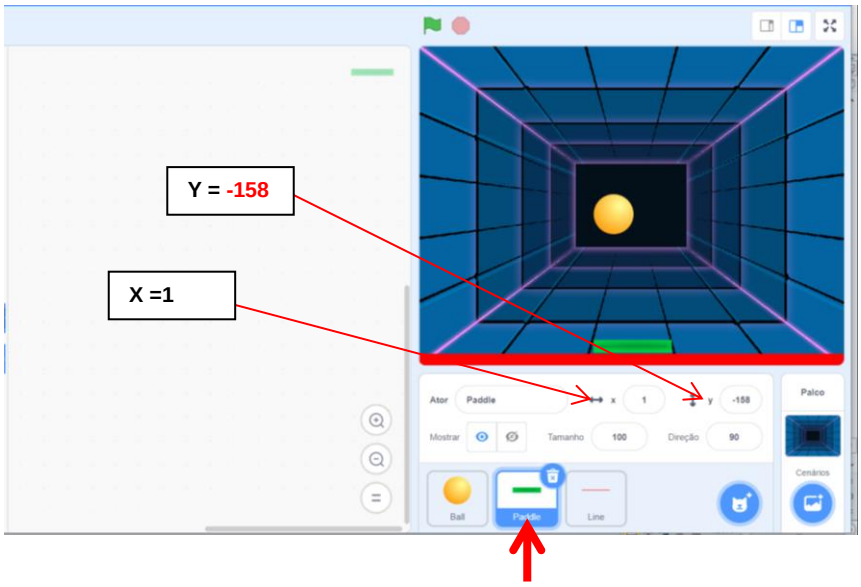


Agora temos três atores e um cenário, lembre-se que quando vamos programar um ator ou cenário **precisamos selecioná-lo na barra inferior**.

Primeiro vamos posicionar o ator LINE:

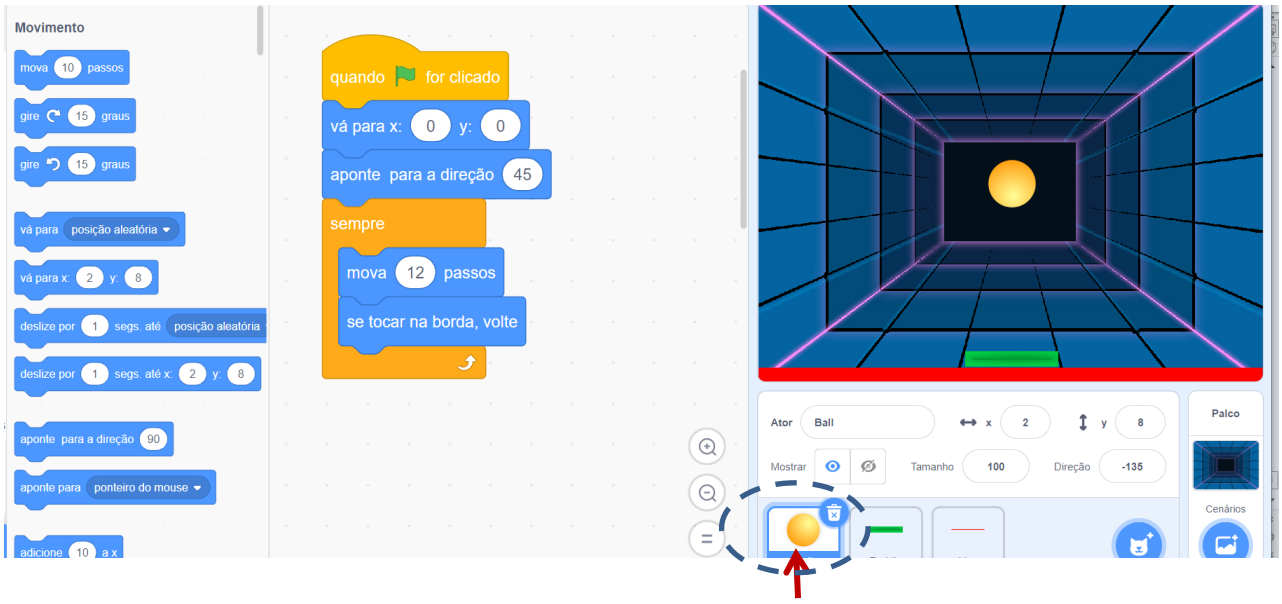


Agora vamos posicionar o ator PADDLE:



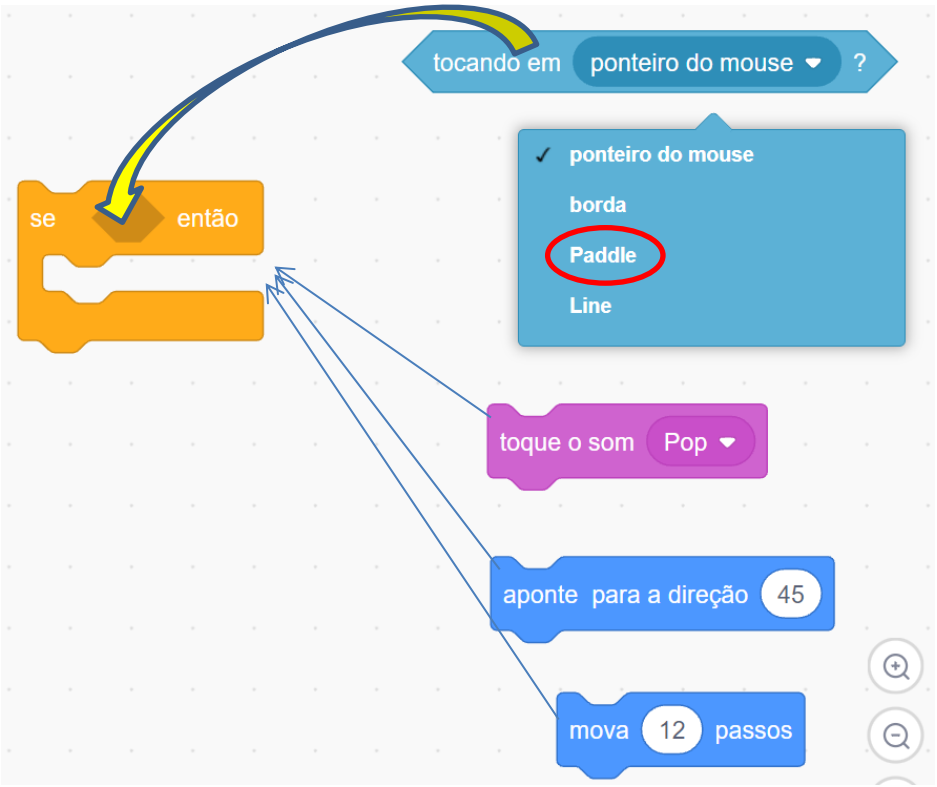
Agora vamos começar a programação. **Primeiro com o ator BALL**, vamos criar três blocos de comando para ele.

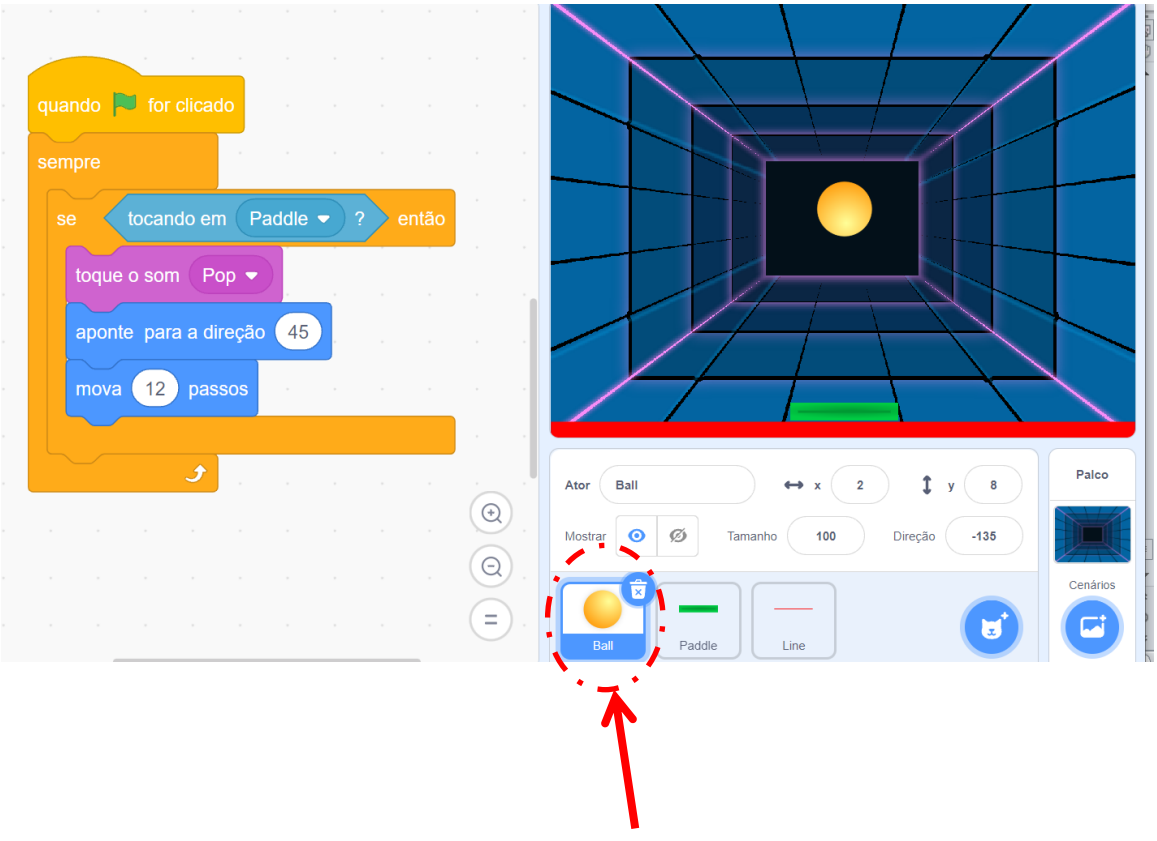
Primeiro bloco: movimentar-se na tela



Segundo bloco: Se tocar no ator PADDLE, continua o movimento.

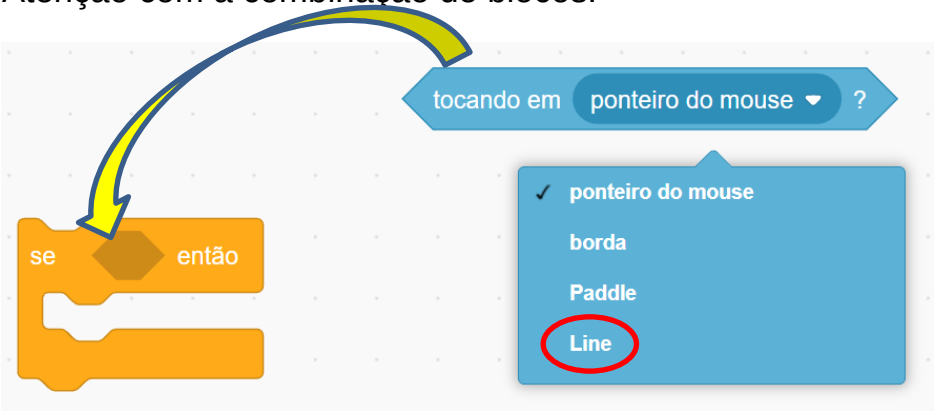
Atenção composição de blocos:

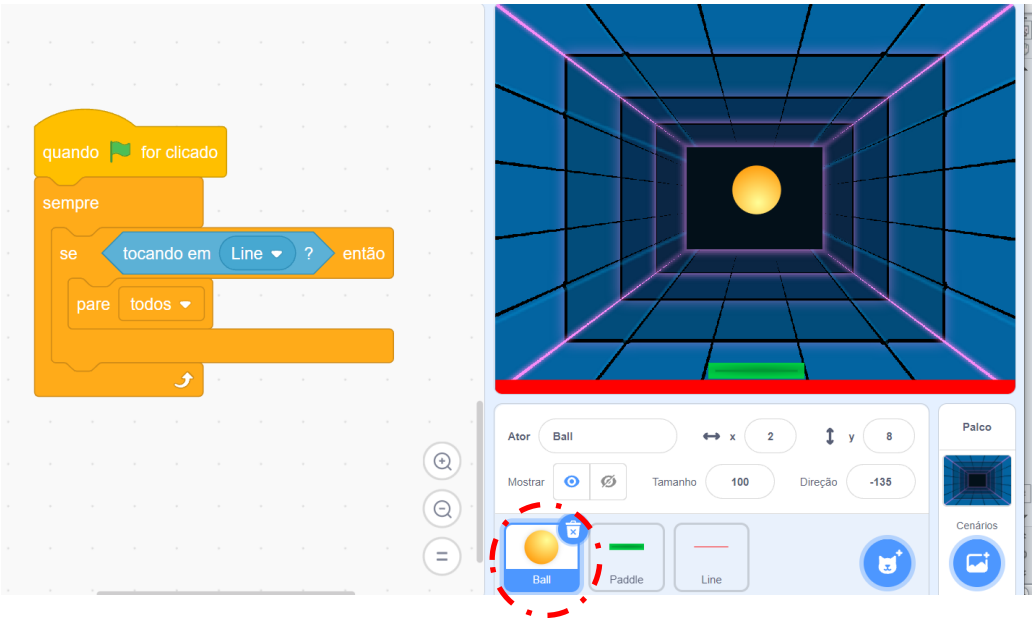




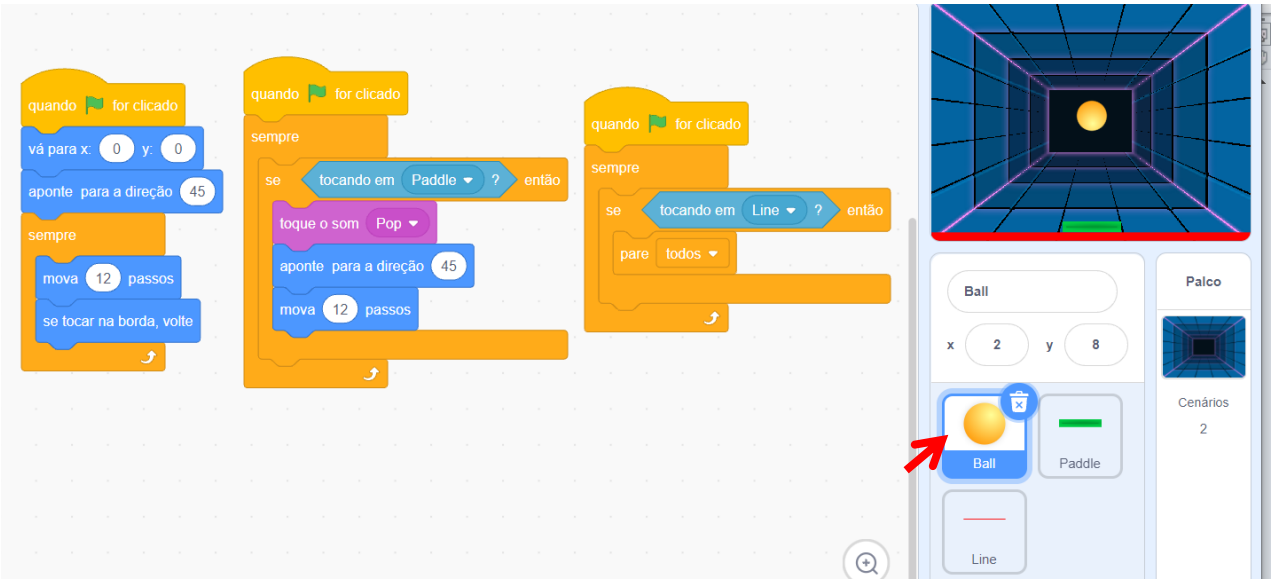
Terceiro bloco: Se tocar no ator LINE, para o jogo.

Atenção com a combinação de blocos:

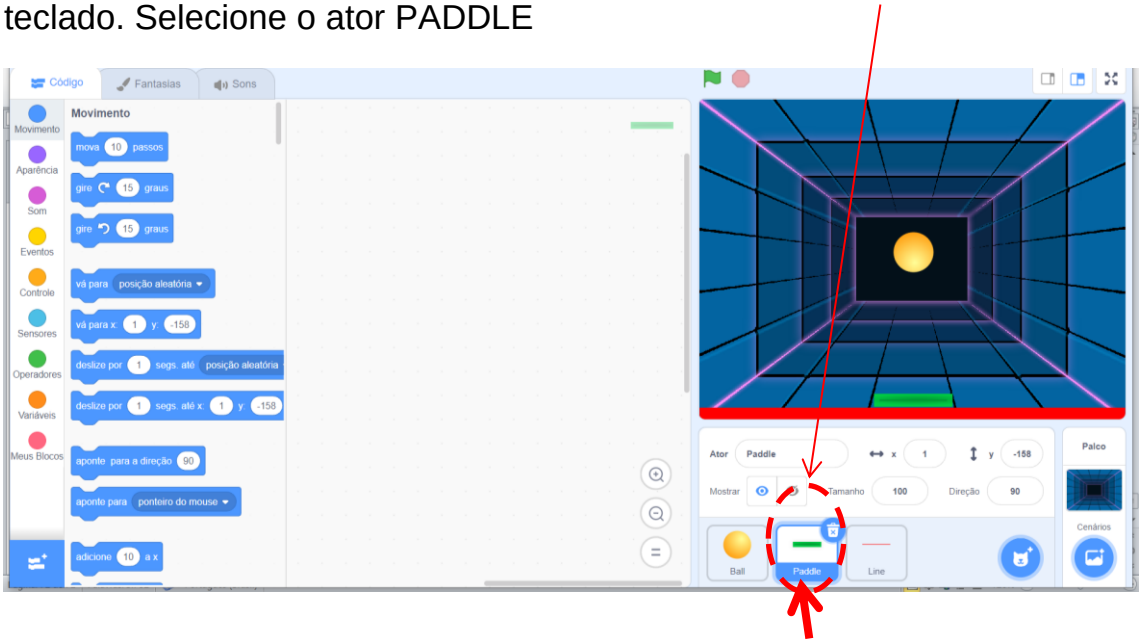




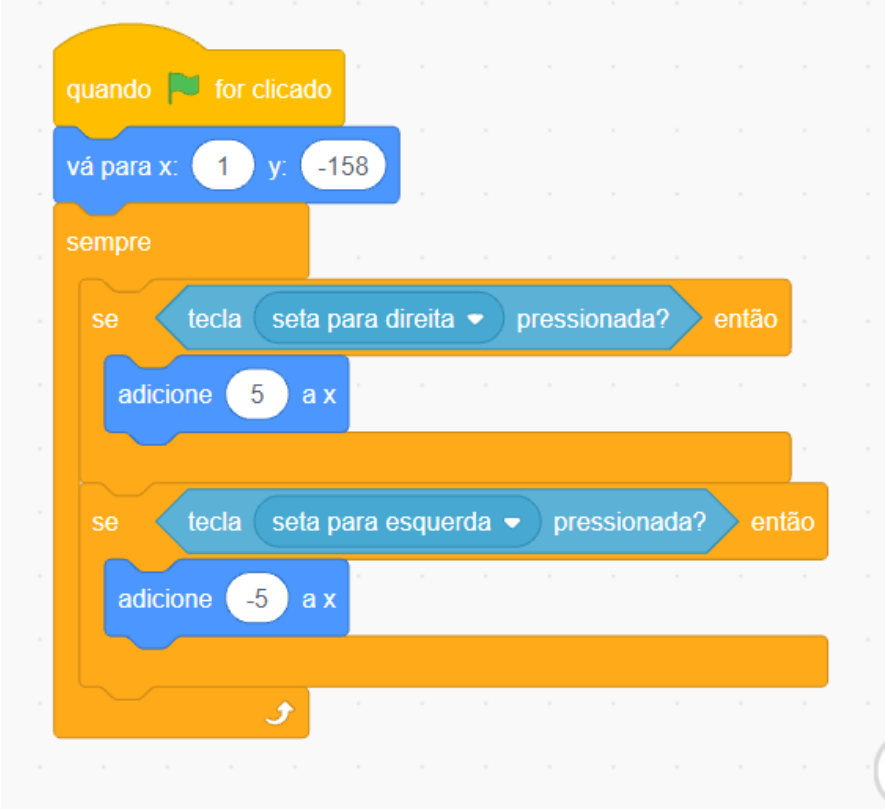
Sua tela deve ficar assim:



Para finalizar, vamos programar o movimento do **ator PADDLE** com o teclado. Selecione o ator PADDLE



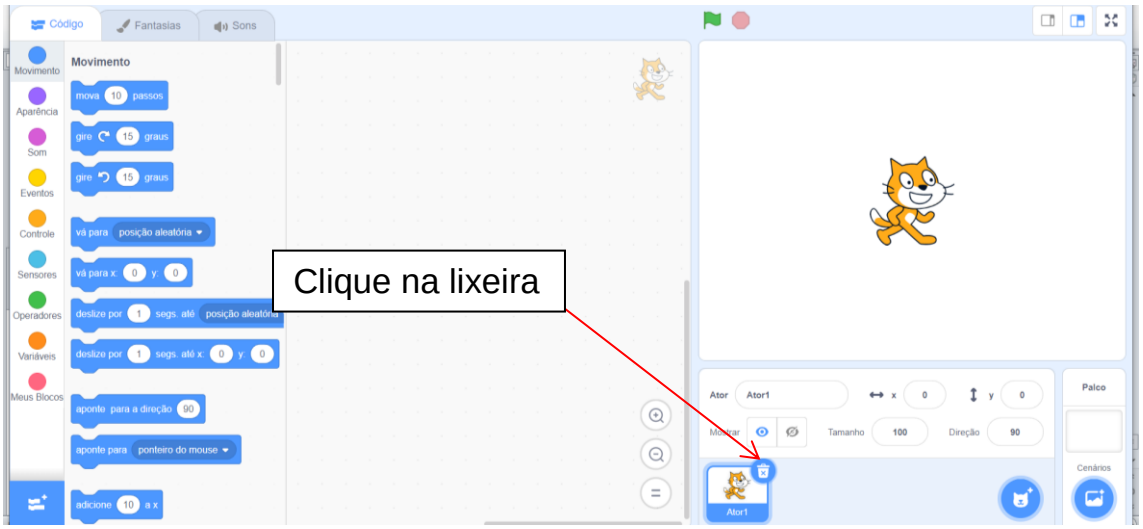
Atenção com a combinação de blocos:



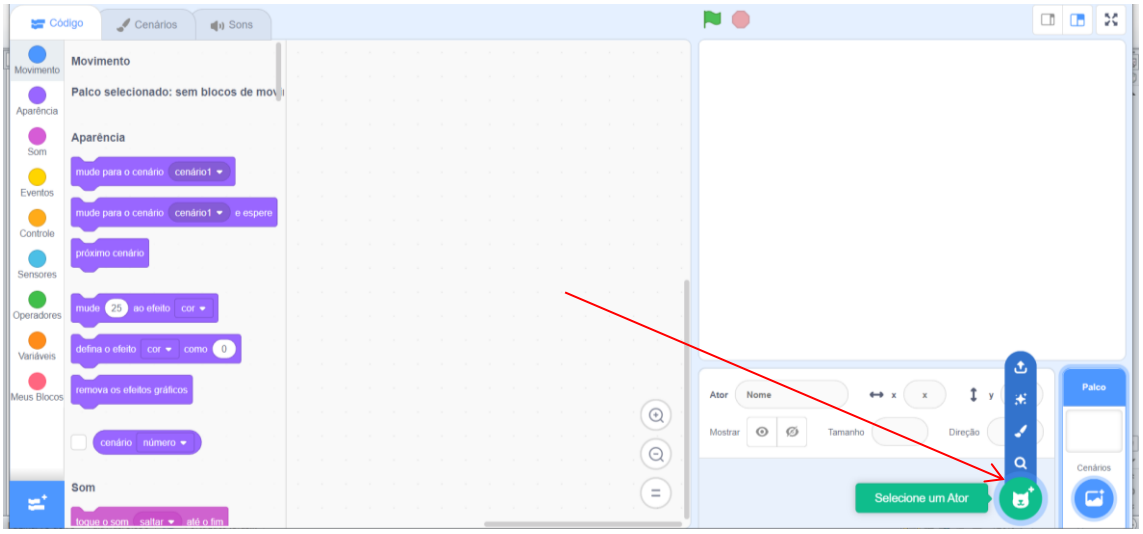
Faça o teste do jogo e verifique se o jogo está funcionando corretamente. Você pode aumentar a dificuldade do jogo, para isso, basta aumentar quantidade de passos no personagem BALL

Aula 10: Esta é nossa última aula do curso. Vamos criar uma animação utilizando o recurso de detecção de vídeo no Scratch e os conceitos das aulas anteriores.

Abra o scratch e exclua o ator inicial:



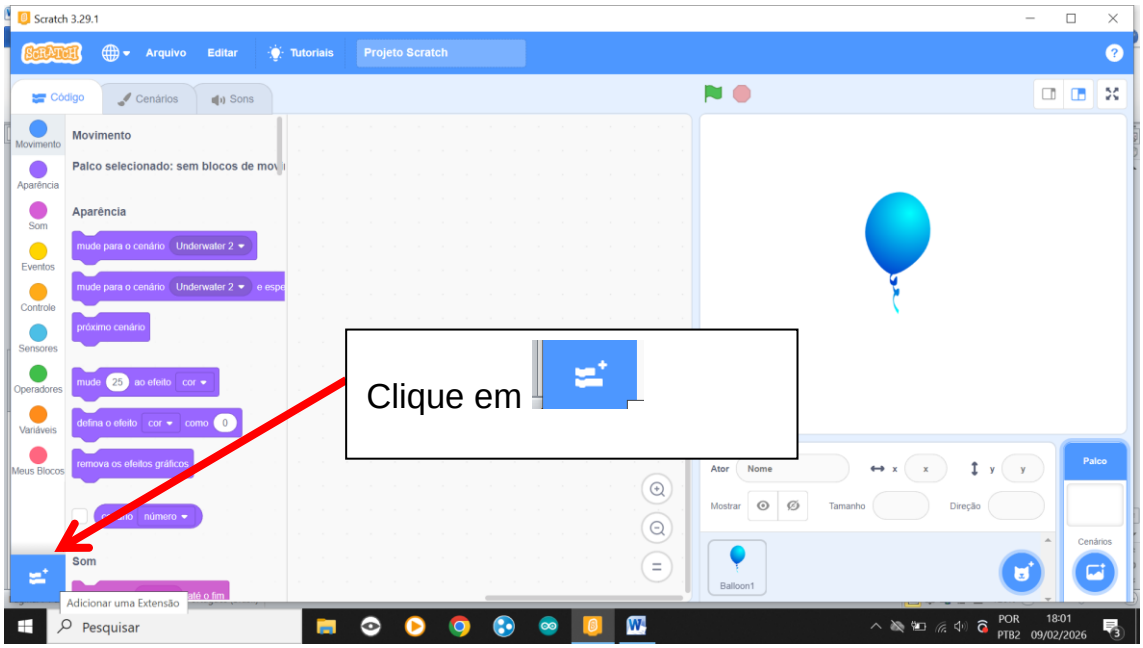
Clique em selecionar ator:



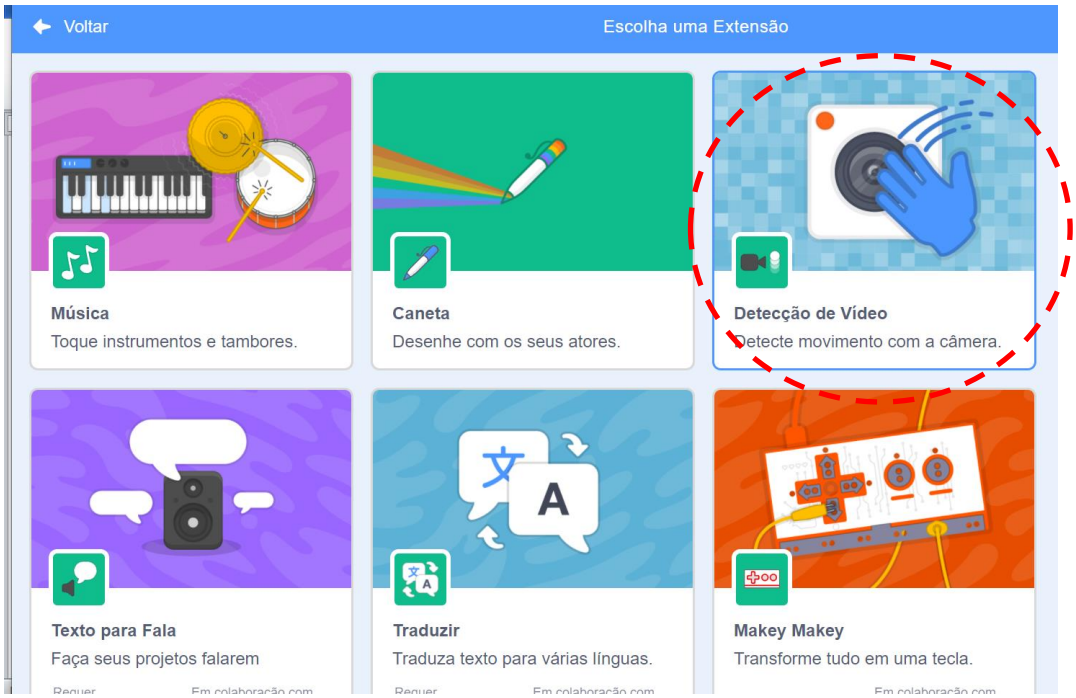
Escolha o ator Ballon 1

A ideia é criar uma animação em que o ator Ballon1 crie clones e fiquem deslizando vertical. Com o recurso da detecção de vídeo vamos movimentar nossas mãos e estourar os balões.

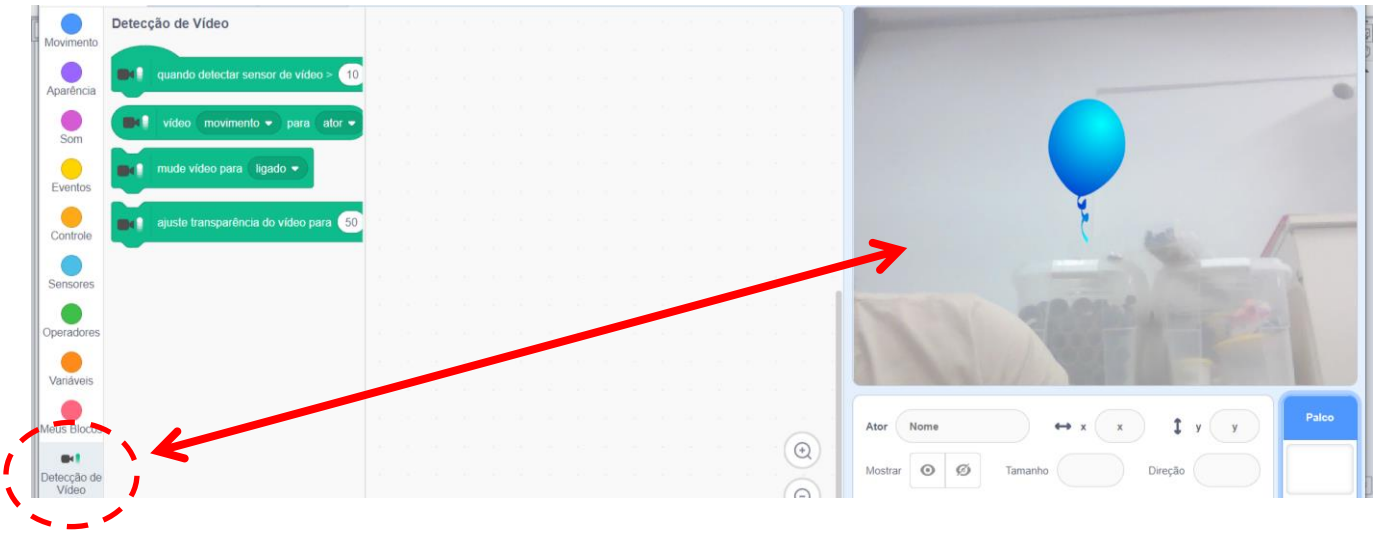
Precisamos habilitar a extensão para detecção de vídeo:



Agora clique em detecção de vídeo:

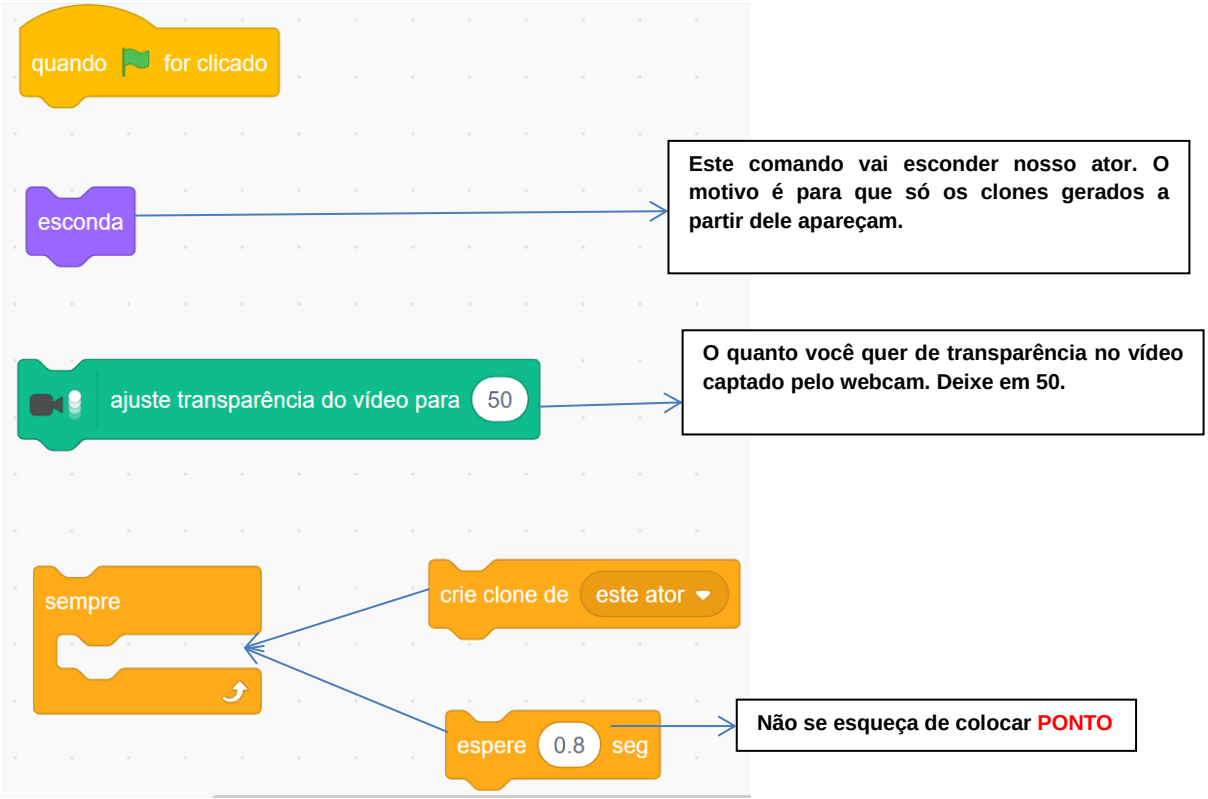


Observe que as ferramentas de “detecção de vídeo” irão aparecer e a webcam irá ligar.



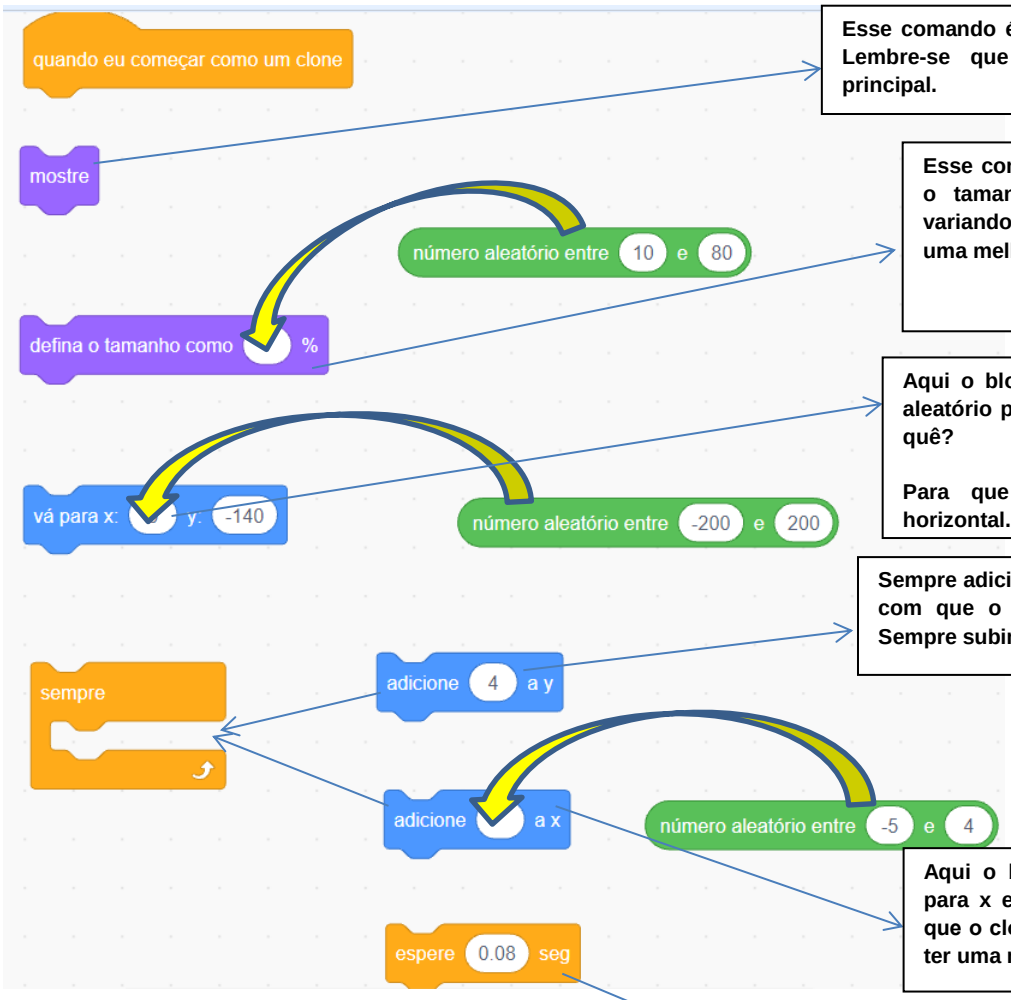
Serão três blocos de programação. Como já mencionado anteriormente em outras aulas, o importante é entender a lógica da programação e **não apenas copiar os blocos.**

Primeiro bloco:



Segundo bloco:

E quando começarem os clones, o queremos que aconteça?



Esse comando é para mostrar o clone. Lembre-se que escondemos o ator principal.

Esse comando é para mostrar aumenta o tamanho do clone. Vamos deixar variando entre 20 e 80. É para o jogo ter uma melhora visual.

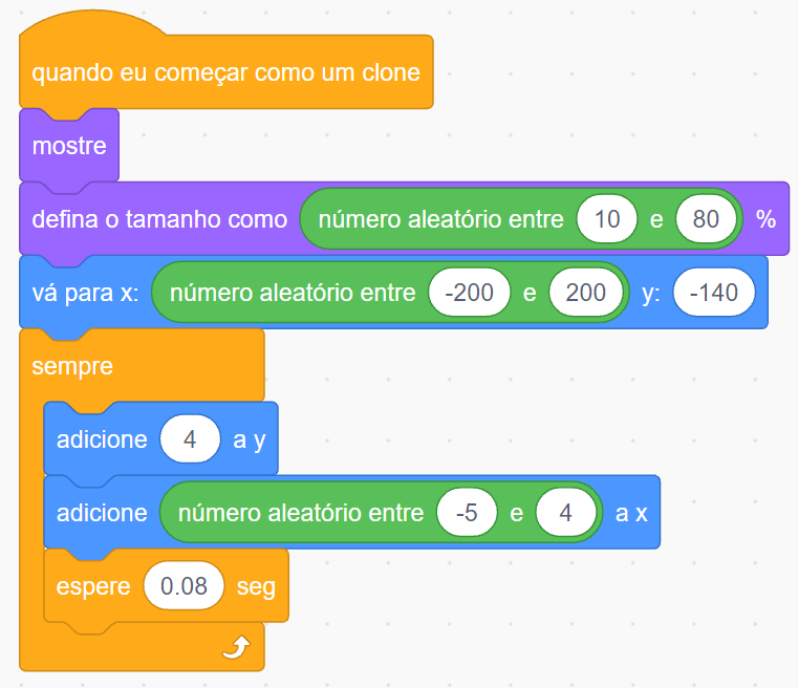
Aqui o bloco vai escolher um número aleatório para o x entre -200 e 200. Por quê?

Para que os clones apareçam no horizontal.

Sempre adicionar 4 na coordenada y faz com que o clone deslize na vertical. Sempre subindo 4 unidades.

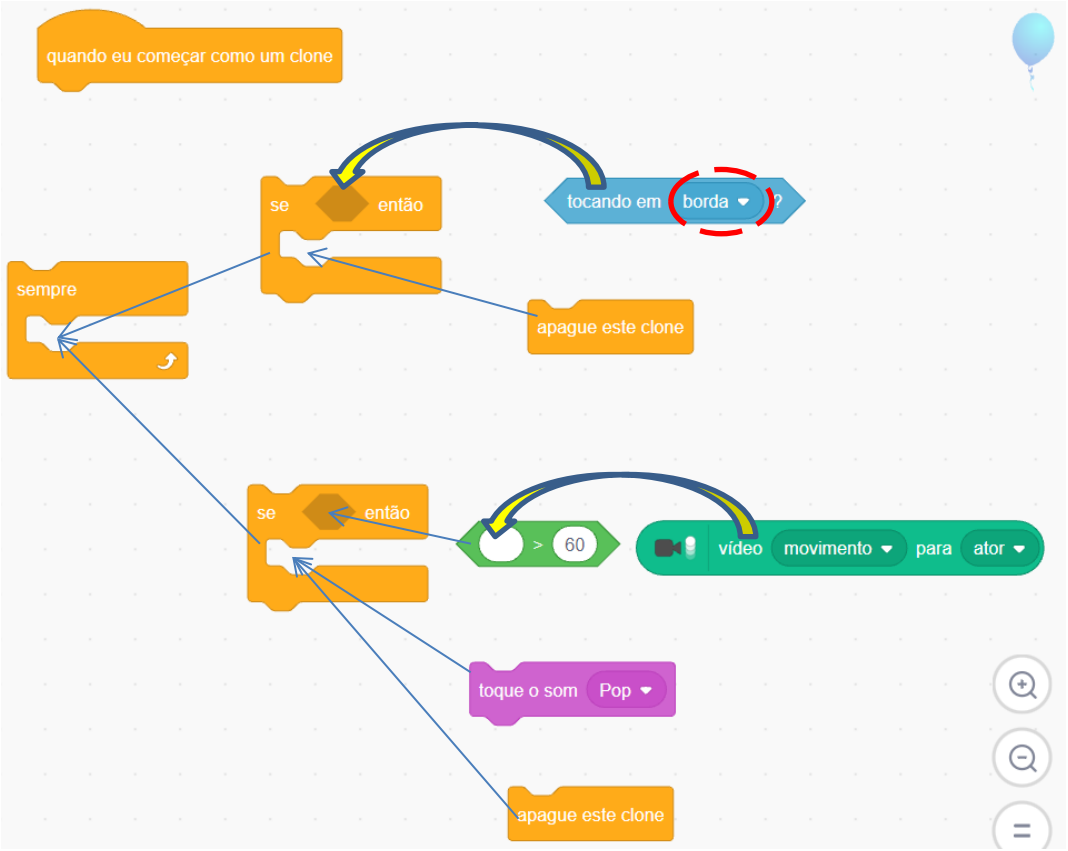
Aqui o bloco vai escolher um número para x entre -5 e 4. Isso vai fazer com que o clone fique "tremendo". É só para ter uma melhora visual do jogo.

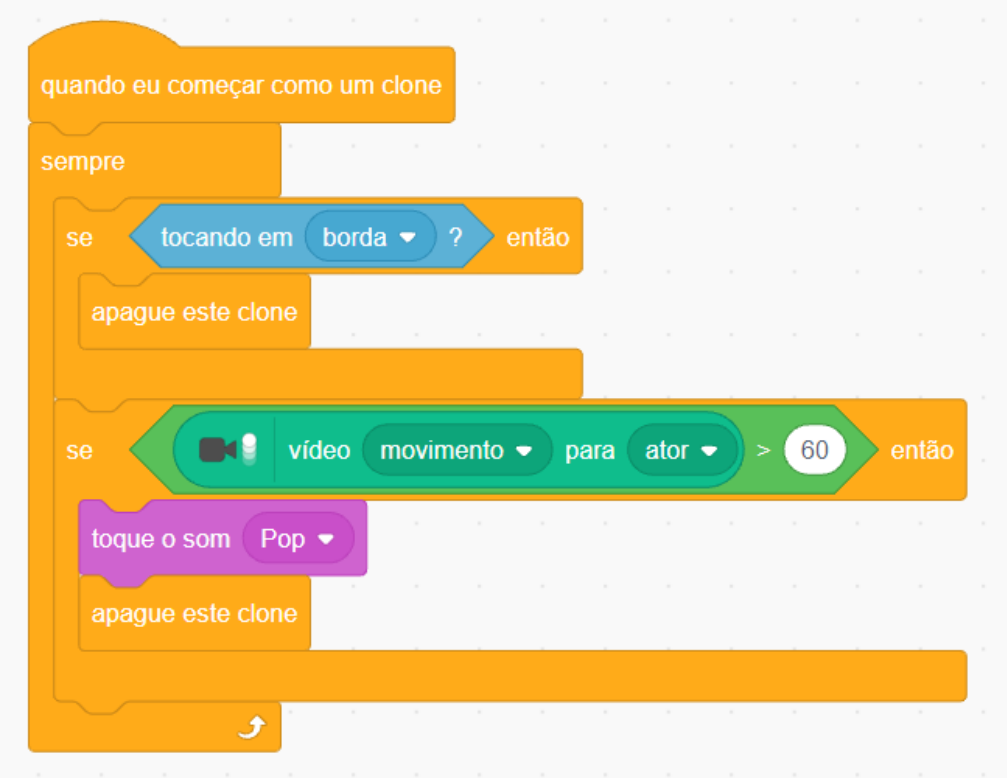
Zero PONTO zero oito



Terceiro bloco:

Agora vamos criar um bloco de comando que vai estourar a bexiga toda vez que a tocarmos com a mão (imagem webcam).





Agora faça um teste e verifique se está funcionando corretamente.